1987

Simulation of automatic control for tractor guidance

Jianlin Ge
Iowa State University

Simulation of automatic control for

tractor guidance

by

Jianlin Ge

A Thesis Submitted to the

Graduate Faculty in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

Major: Agricultural Engineering

Iowa State University
Ames, Iowa

1987

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

Directional stability is one of the basic problems in vehicle handling. It relates to the ability of a vehicle to stabilize its direction of motion against external disturbances such as the lateral forces generated by crosswinds, weight transfer of the vehicle, turning maneuvers, etc. Therefore, to evaluate the directional stability of the vehicle, it is important to consider the lateral motion of the vehicle as a time response when the vehicles is subjected to lateral disturbances. The study of the directional stability had its origins in aeronautical engineering. Later on, with the rapid development of the automobile industry, the directional stability of road vehicles was taken more seriously by both drivers and manufacturers, due to the great increase in vehicle speed. So the research into directional stability was performed by the automobile industry in order to improve the handling response of the road vehicles.

Recently, much research has been done on the directional stability of the road vehicles (Huston and Johnson, 1982). With the development of both digital and analog computers, the analysis and simulation of more complicated multi-degree models becomes possible and more information on the directional dynamics can easily be obtained. Compared with the research done on the directional stability of road vehicles, little effort has been made on off-road vehicles, especially on agricultural tractors. Although tractors are relatively slow in forward speed, the working conditions for them are more complicated than those for the road vehicles. Directional stability, however, is generally required to keep the tractors or tractor-implement combinations on prescribed paths.

In this paper, the lateral transient behavior of a particular four-wheeled tractor under various kinds of disturbances will be analyzed by using a bicycle model. The transient response of the tractor will be obtained to investigate the directional stability of the tractor.

The driver and the tractor characteristics both contribute to the directional stability of the tractor. It is the driver that provides the control signals such as step, ramp or random signals at proper times according to the particular environmental circumstances. One section of this paper will deal with the lateral motion of the tractor and the dynamics of the driver-tractor model. Using results from research and experiment, Carson and Wierwille (1978) indicated that human operators have a certain delay between the time an error occurs in a system output and the time that a corrective action is taken. Also inherent in the human operator is a finite threshold to the error amplitude and frequency. Besides, some field operations require that a system follows a precise course and that is beyond the capability of the human operators.

The objective of this paper is to simulate the directional behavior of the four-wheeled agricultural tractor and enhance the directional stability of the tractor by means of PID (Proportional-Integral-Derivative) automatic control.

The specific objectives of this study were:

1. to develop a simplified mathematical model of a four-wheeled agricultural tractor and analyze its lateral motion behavior under operation on an unyielding horizontal surface.
2. to introduce a conventional PID automatic controller into the tractor for the yaw-angle control.
3. to develop a computer program to simulate the yaw angle of the tractor when subjected to different external lateral disturbing inputs and select the 'optimal' control gains that realize satisfactory directional stability of the tractor.

Based on the classical bicycle model (Moldenhauer, 1985), the Laplace transform was used in developing and analyzing the motion of the system. Digital computer simulation is applied in the study of the time response of the tractor and in the selection of the PID controller. Graphical representations of the time response were made to interpret the control gains selected by the optimization program.

LITERATURE REVIEW

The literature reviewed here can be generally divided into three parts: the tractor, the driver and the automatic controller.

The Vehicle

In the past decades, considerable research effort has been expended on analyzing dynamic behavior of automobiles, but, at the outset, the work was restricted to qualitative evaluation of experimental measurements of vehicle motion (Stonex, 1941). It was after World War II that the analytical methods used in aeronautical engineering began being transferred and gradually applied to the automobile industry. Schilling (1953) was among the first to apply theoretical methods in the analysis of the dynamics of the road vehicles and in the solution of the problems in the directional stability of the road vehicles.

Since the 1930's, automobile speed has increased considerably due to the improvement of the vehicle engine and to road conditions. As a consequence, the problem of the stability of the road vehicles has become more serious and has led to numerous studies on vehicle handling. Evans (1935) proposed that the most important factor that affected vehicle handling was the tire. Such properties as cornering stiffness and slip angle were expressed in quantitative terms. In dealing with the cornering characteristics of the tires and their effect on the lateral motion of the vehicle, Brouhleit (1925) seems to have been the first person to introduce the force-slip angle concept in the analysis of the lateral motion of the vehicles.

Directional stability is one of the vehicle handling characteristics and, to a great extent, determines the lateral stability of the vehicles in the transient state. In the study of the directional dynamics of the vehicles under external disturbances, much contribution was made by Bundorf and Pollock (1963).

The analysis of the lateral stability of the vehicles at first was based on the simple, linear and three-degree bicycle model due to its simple analysis and easy verification. Huston and Johnson (1982) have done much work on the analysis of the lateral stability of both three-wheeled and four-wheeled road vehicles by using such kind of linearized bicycle model.

Later on, with the development of both digital and analog computers, more complicated multi-degree nonlinear models were generated and applied in the computer simulation of the lateral motion of the road vehicles (Chiesa and Rinonapoli, 1967).

Recently studies of agricultural tractor stability have been conducted due to the continuous increase in the number of tractor accidents occurring each each year. In his paper, Larson (1971) analyzed the dynamic motion of a four-wheeled tractor moving on the sideslopes by using a three degree of freedom model. Further such research has been done by Denny and Rehkugler (1974) in mathematical modelling, computer simulation, and model verification of wheel-tractor overturns. A ten-degree mathematical model has been developed to describe large-magnitude motions of wide-front-end wheel tractors traversing on either smooth or irregular ground surfaces and to predict the tractor overturn motions.

Although the multi-degree models can be simulated by more powerful computers and more complete dynamic behavior of the vehicle can be obtained theoretically, it is difficult to verify those complex models practically. In the Ph.D. dissertation presented by Moldenhauer (1985), a classical bicycle model was utilized in the theoretical analysis of the lateral stability of road vehicles and many basic dynamic characteristics of the vehicles can be well obtained. For this reason, the classical bicycle model will be introduced in this thesis for the simulation of the lateral motion of a four-wheel agricultural tractor and for the problem of the directional stability of the tractor when subjected to external disturbances.

The Driver

A conventional automobile does not have the intelligence to overcome external disturbances or to perform other prescribed maneuvers without the guidance of the driver. It is the driver's ultimate responsibility to provide appropriate control inputs to maintain the lane position of the vehicle.

In dealing with the dynamics of the driver/vehicle system, both simple and complex models for the driver have been developed. In general, these models can be categorized as: linear model, such as proportional; quasi-linear, predictive and optimal control models; and nonlinear model, such as a strategy model.

The quasi-linear model is a double-loop compensatory model which represents marginal response of the driver physiologically. An excellent summary of this kind of model is given by McRuer and Krendel (1974). The predictive model is of the single-loop type, which is characterized by an input cue that comes from the difference between the set point and the driver's sight point. Kondo and Ajimine (1968) have done much research in this aspect. The optimal control model was first developed for the aircraft simulation. The key idea underlying this model is the assumption of the nearly optimal behavior of the operator with time delay and observational noise. In the paper presented by MacAdam and Fancher (1986), the optimal control model is studied in detail.

Based on the characteristics of the driver, such as his finite threshold to the error amplitude and frequency and the time delay in reaction, the strategy model is developed. This is an open-loop control model with the lateral motion state variables of a vehicle as inputs and the steering wheel position as output (Carson and Wierwille, 1978).

For the agricultural tractors it may be assumed that tractors are simple automobiles without any suspension system, with low constant forward speed, small slip angle and negligible rolling motion. Thus, in developing the driver/tractor model, the driver can be simplified as a proportional controller with unit feedback. The analysis of the driver dynamic properties and the simulation of the directional response of the driver/tractor system will be based on this proportional driver model and the bicycle vehicle model.

The Automatic Controller

In practice, the driver as a proportional controller is unable to perform each control command accurately under a complex ground condition. As mentioned previously, there are three characteristics of a human driver: response includes a time lag, there is some minimum error from set point that is detectable, and a human has a frequency response limit. Physical fatigue, which is caused by the constant concentration required to hold the tractor on the prescribed path, affects the driver's capability for the directional control of the system. The idea of keeping the system precisely in the right position and orientation, and the challenge of reducing the driver's fatigue associated with wearisome field operations result in the development of the automatic controller for the tractor guidance.

At the beginning of the century, the control theory and its practice in the United States of America was mainly applied to telephone systems for electrical signal control (Bode, 1964). The large impetus to the theory and practice of automatic control occurred during World War II when it became necessary to design and construct feedback controllers for military systems such as airplanes and radars (Thaler, 1974). In recent years, an increasing research effort has been devoted to automatic guidance of both road and off-road vehicles. Physically, automatic controllers can be categorized as analog or digital controllers. The error between the set point and the output is usually fed back to the controlling element in the plant after amplification either directly (proportional), or after integration (integral), or after differentiation (derivative). A combination of all three forms of feedback is common (Palm, 1983).

The application of automatic controllers in agricultural tractors can be found in the area related to the guidance of the tractors (Schafer and Young, 1979). A conventional 3-mode PID controller will be proposed for tractor guidance in this thesis.

The analysis of the lateral directional dynamics of a real system and the prediction of its time response are often most difficult and time consuming due to the interrelationship among variables and the complex interaction of related parameters. Although analytical solutions are not often attainable, numerical solutions can be obtained by using a digital computer.

As early as 1963, Ellis (1963) developed a three degree of freedom analog computer model for the study of the lateral motion of an articulated vehicle. Then, with the advent of more sophisticated computers, the simulation of multi-degree of freedom, nonlinear vehicle models has been simplified (Speckhart, 1973). Recently, the application of the computer simulation has penetrated into the agricultural engineering for the study of the lateral stability of tractors (Rehkugler, 1980; Xie, 1984).

Digital computer simulation will be applied in this thesis for the study of the lateral motion behavior of a PID-controller/tractor system and the characteristics of the control gains that govern the directional stability of the system.

MATHEMATICAL MODEL FOR FOUR-WHEEL TRACTORS

Most published papers on vehicle stability have discussed the stability of cars and trucks at highway speeds (Huston and Johnson, 1982; Moldenhauer, 1985). Overturning through rolling about the longitudinal center line is a possible outcome of attempting a curve at too great a speed. Although agricultural vehicles are subjected to overturning, such behavior is almost always the result of attempting to travel on a steep slope (Larson, 1971; Rehkugler, 1980). In this thesis the tractor will be considered to move at 5 m/s or less on a hard and level surface. Only small deviations from a straight-ahead path will be considered because this thesis is concerned with automatic guidance to maintain the tractor on a preset path. Recognizing these constraints, a two dimensional model has been chosen to simplify this initial investigation of automatic guidance.

Owing to the fact that just the lateral motion variables, such as lateral velocity and yaw rate, are involved in the study of the lateral directional dynamics of the tractor, a three degree of freedom bicycle model in two dimensions is adequate for developing the dynamic equations of motion and illustrating the basic dynamic characteristics of the tractor. The bicycle model is a classical model with zero width and lumped wheels on the longitudinal center line of the tractor. The bicycle model used for the study of the lateral motion of the four-wheeled, rear-drive tractor is shown in Fig. 1.

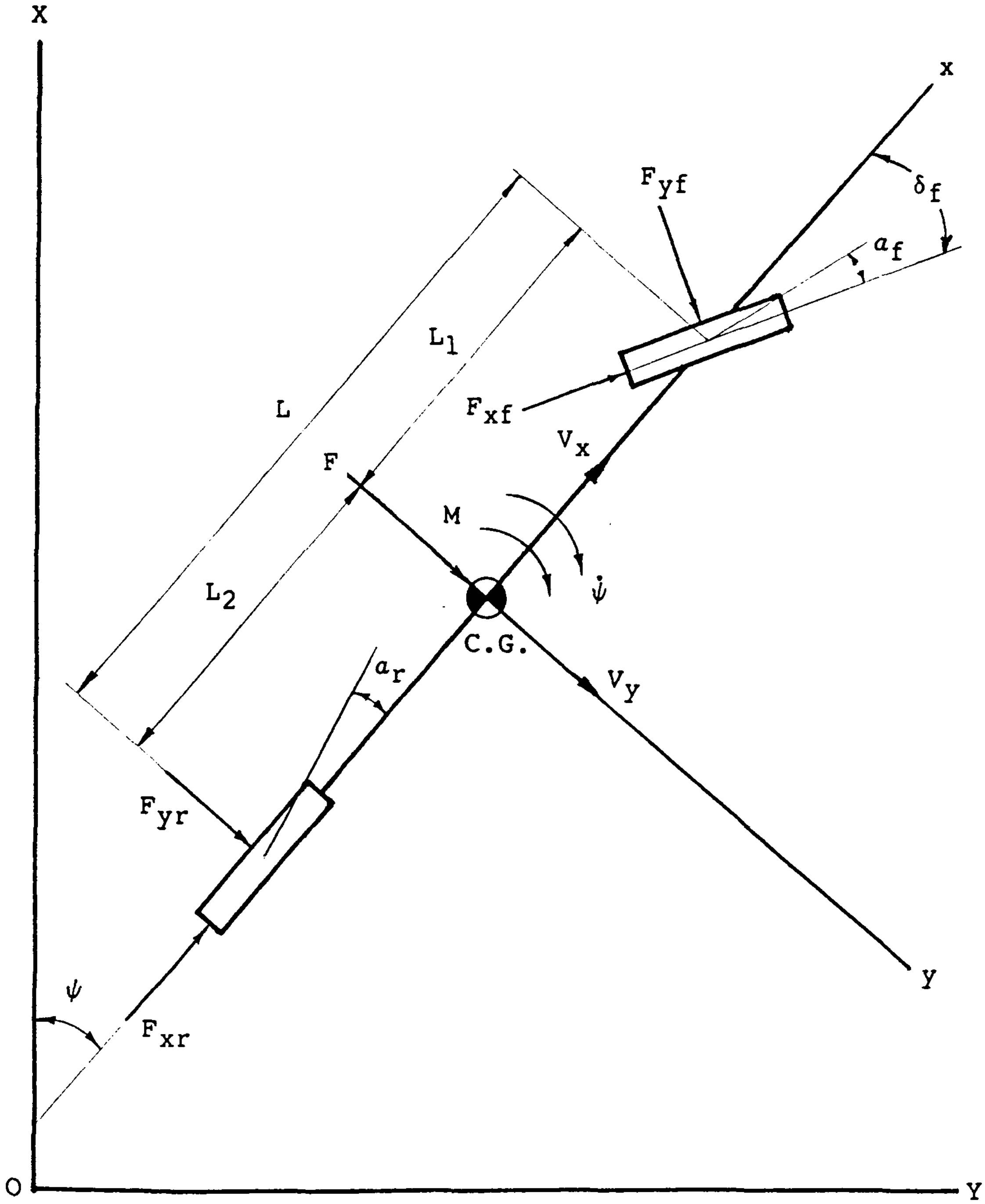


FIGURE 1. Bicycle model of the tractor

A local right-hand set of Cartesian axes xoy is selected and fixed at the center of gravity of the tractor in order to derive the rate variables (lateral velocity and yaw velocity) and to simplify the motion analysis. In order to define the absolute position and the orientation (lateral displacement and yaw angle) of the tractor, an inertia reference frame XOY is also introduced and fixed in space.

The wheels on each axle are lumped to the center line of the tractor with zero width of the model. The front axle is located a distance L_1 in front of the center of gravity (C.G.) and the rear axle is L_2 behind the C.G. of the tractor.

The following basic assumptions are made in the development of the mathematical model:

1. The ground surface upon which the wheels travel is a hard, smooth and flat plane.
2. The forward speed of the tractor is less than 5 m/s and constant.
3. The rolling motion of the tractor is negligible.
4. The lateral load transfer of the tractor on the tires is negligibly small.
5. The relationship between slip angle and lateral force for the tires is linear.
6. The steer angle of the front wheels is small.
7. The tractor is a rear-wheel drive tractor.
8. The tractor is a simple vehicle with unsprung mass.
9. Draught force is not considered.

Newtonian rigid-body dynamics is applied in the derivation of the motion equations that govern the lateral motion behavior of the tractor.

Suppose that the tractor is moving at a constant speed along a straight line path when it is subjected to external disturbances, a force F and a moment M. The tractor will be in a transient state for a certain period of time before returning to the steady state. The lateral motion behavior of the tractor, to a great extent, determines the directional stability of the tractor system.

The dynamic equation of motion of the system in x direction is obtained by summing all the external forces along the x-axis:

$$ma_x=m(\dot{V}_x-V_y\Omega_z)=F_{xf}\cos\delta_f+F_{xr}-F_{yf}\sin\delta_f \tag{1}$$

The translational equation of motion in y direction is defined by summing all the external forces along the y-axis:

$$ma_y=m(\dot{V}_y+V_x\Omega_z)=F_{yf}\cos\delta_f+F_{yr}+F_{xf}\sin\delta_f+F \tag{2}$$

The rotational equation of motion is expressed in terms of the yaw angular acceleration of the system about z axis through the C.G. of the system and perpendicular to xoy plane:

$$I_z\dot{\Omega}_z=I_z\ddot{\psi}=L_1F_{xf}\sin\delta_f+L_1F_{yf}\cos\delta_f-L_2F_{yr}+M \tag{3}$$

where, a_x = longitudinal acceleration,

a_y = lateral acceleration,

F_{xf} = front-wheel tractive force,

F_{xr} = rear-wheel tractive force,

F_{yf} = front cornering force,

F_{yr} = rear cornering force,

I_z = moment of inertia about z axis,

m = mass of the tractor,

Ω_z = yaw rate of the tractor,

V_x = longitudinal velocity of the tractor in xoy coordinate system,

V_y = lateral velocity of the tractor in xoy coordinate system,

δ_f = front steer angle,

F = lateral disturbing force acting at the C.G.,

M = external disturbing moment about z axis through the C.G..

Experimental and analytical investigations of the cornering characteristics of pneumatic tires indicate that a linear relationship exists between the cornering force and the slip angle, if the slip angle is below 4° (Wong, 1978). For the four-wheel tractor, the following expressions hold:

$$F_{yf} = 2C_{\alpha f}\alpha_f \tag{4}$$

$$F_{yr} = 2C_{\alpha r}\alpha_r \tag{5}$$

where, α_f = front slip angle,

α_r = rear slip angle,

$C_{\alpha f}$ = front cornering stiffness,

$C_{\alpha r}$ = rear cornering stiffness.

Supposing that the slip angles of both front and rear tires are sufficiently small, then the following approximations of the slip angles are reasonably true:

$$\delta_f-\alpha_f \simeq \tan(\delta_f-\alpha_f)=(V_y+L_1\Omega_z)/V_x \tag{6}$$

$$\alpha_r \simeq \tan\alpha_r=(V_y-L_2\Omega_z)/V_x \tag{7}$$

If further assumptions are made of small front steer angle, rear-wheel drive and constant longitudinal speed, by substituting Eqs. (4)~(7) into Eqs. (1), (2) and (3), Eq. (1) vanishes. Thus, the dynamic equations of motion governing the directional dynamics of the system are reduced to two:

$$m\dot{V}_y+a_1\Omega_z+a_2V_y=2C_{\alpha f}\delta_f+F \tag{8}$$

$$I_z\dot{\Omega}_z+a_3\Omega_z+a_4V_y=2L_1C_{\alpha f}\delta_f+M \tag{9}$$

where, $a_1 = mV_x+(2L_1C_{\alpha f}-2L_2C_{\alpha r})/V_x$,

$a_2 = (2C_{\alpha f}+2C_{\alpha r})/V_x$,

$a_3 = (2L^2{}_1C_{\alpha f}+2L^2{}_2C_{\alpha r})/V_x$,

$a_4 = (2L_1C_{\alpha f}-2L_2C_{\alpha r})/V_x$.

In order to simplify the evaluation of the differential equations (8) and (9), the Laplace transform will be introduced in subsequent sections for the study of the dynamic motion of the system. Also the related block diagram will be developed in conjunction with the Laplace equations in presenting a graphical relationship among input, intermediate, and output variables.

By taking the Laplace transform with the initial conditions $V_y(0) = 0$ and $\Omega_z(0) = 0$, the system of the linear differential equations (8) and (9) becomes the system of algebraic equations below:

$$mSV_y(S)+a_1\Omega_z(S)+a_2V_y(S)=2C_{\alpha f}\delta_f(S)+F(S) \tag{10}$$

$$I_zS\Omega_z(S)+a_3\Omega_z(S)+a_4V_y(S)=2L_1C_{\alpha f}\delta_f(S)+M(S) \tag{11}$$

Rearranging Eqs. (10) and (11), the system of equations becomes:

$$(mS+a_2)V_y(S)+a_1\Omega_z(S)=2C_{\alpha f}\delta_f(S)+F(S) \tag{12}$$

$$a_4V_y(S)+(I_zS+a_3)\Omega_z(S)=2C_{\alpha f}L_1\delta_f(S)+M(S) \tag{13}$$

Assume that the dynamic system is under fixed control and is traveling along a straight line ($\delta_f = 0$), when subjected to impulse external disturbances F(S) and M(S). The solutions of the rate variables in S-domain is obtained by solving Eqs. (12) and (13):

$$V_y(S)=(I_zF(S)S+a_3F(S)-a_1M(S))/((mS+a_2)(I_zS+a_3)-a_1a_4) \tag{14}$$

$$\Omega_z(S)=(mM(S)S+a_2M(S)-a_4F(S))/((mS+a_2)(I_zS+a_3)-a_1a_4) \tag{15}$$

Further simplify Eqs. (14) and (15),

$$V_y(S)=A(S+P_1)/(S^2+2\zeta\omega_nS+\omega^2{}_n) \tag{16}$$

$$\Omega_z(S)=B(S+P_2)/(S^2+2\zeta\omega_nS+\omega^2{}_n) \tag{17}$$

where, $A = F(S)/m$,

$B = M(S)/I_z$,

$P_1 = (a_3F(S)-a_1M(S))/I_zF(S)$,

$P_2 = (-a_4F(S)+a_2M(S))/I_zF(S)$,

$\zeta = (a_2I_z+a_3m)/2((a_2a_3-a_1a_4)mI_z)^{1/2}$,

$\omega_n = ((a_2a_3-a_1a_4)/mI_z)^{1/2}$.

From Eqs. (16) and (17) it can be seen that the characteristic equation of the dynamic system is:

$$S^2+2\zeta\omega_nS+\omega^2_n=0 \tag{18}$$

According to the Routh-Hurwitz criterion, the necessary and sufficient condition for a stable second order system is that the coefficients of Eq. (18) must all be positive. The conditions that should be met for the stable system are:

$\omega^2_n=(a_2a_3-a_1a_4)/mI_z > 0$ and

$\zeta > 0$,

or $a_2a_3-a_1a_4 > 0$

Through simple algebraic manipulation, finally the following inequality is obtained:

$$V_x < (gL/K_{us})^{1/2} \tag{19}$$

where, g = acceleration of gravity,

L = the wheel base of the tractor,

K_{us} = understeer coefficient

$= (W_f/2C_{af})-(W_r/2C_{ar})$,

W_f = normal load of front axle,

W_r = normal load of rear axle.

The Eq. (19) indicates that in order to keep the dynamic system under the stable condition the forward speed V_x of the tractor should be less than a critical speed, namely,

$$V_{crit}=(gL/K_{us})^{1/2}$$

For an agricultural tractor, usually $K_{us} > 0$ (Wong, 1978). Furthermore, for the field working tractor concerned with in this paper, the working speed is relatively low compared with the road vehicle, and the forward speed V_x usually meets the requirement proposed by the Eq. (19). Thus the rolling of the field-working tractor is not considered as a serious problem.

Based on the Eqs. (16), (17) and (18), the directional stability of the system is studied by the variation of the damping ratio ζ.

If $\zeta \geq 1$, Eq. (18) has two real negative roots. By factorizing the denominator of the Eqs. (16) and (17), we obtain

$$V_y=A(S+P_1)/(S+a)(S+b) \tag{20}$$

$$\Omega_z=B(S+P_2)/(S+a)(S+b) \tag{21}$$

where, $a = \zeta\omega_n+\omega_n(\zeta^2-1)^{1/2}$,

$b = \zeta\omega_n-\omega_n(\zeta^2-1)^{1/2}$.

By inverting the Laplace transform, the time responses of the system in terms of lateral velocity V_y and the yaw velocity Ω_z in Eqs. (20) and (21) are obtained as following:

$$V_y=A((P_1-a)e^{-at}-(P_1-b)e^{-bt})/(b-a) \tag{22}$$

$$\Omega_z=B((P_2-a)e^{-at}-(P_2-b)e^{-bt})/(b-a) \tag{23}$$

The Eqs. (22) and (23) indicate that both lateral velocity V_y and yaw velocity Ω_z will die out within a certain period of time. As shown in Fig. 2, due to the positive real values of coefficients a and b, the attenuation will follow smooth impulse curves without any oscillations.

If $\zeta < 1$, both a and b are a pair of complex conjugate with the real parts positive, that is,

$$a=\zeta\omega_n+(\omega_n(1-\zeta^2)^{1/2})j,$$

$$b=\zeta\omega_n-(\omega_n(1-\zeta^2)^{1/2})j.$$

When $\zeta < 1$ in the time response equations, besides exponential term $e^{-\zeta\omega_n t}$ there is also a term $\sin(\omega_n(1-\zeta^2)^{1/2}t+\phi)$. Although the transient response will vanish after certain time, the system will oscillate due to the existence of the periodic function. It is interesting to note that the oscillation will increase with the decrease of the damping ratio ζ, which might lead to a system that would be difficult to drive. On the contrary, the larger the damping ratio ζ, the smoother and more sluggish the response curves of the

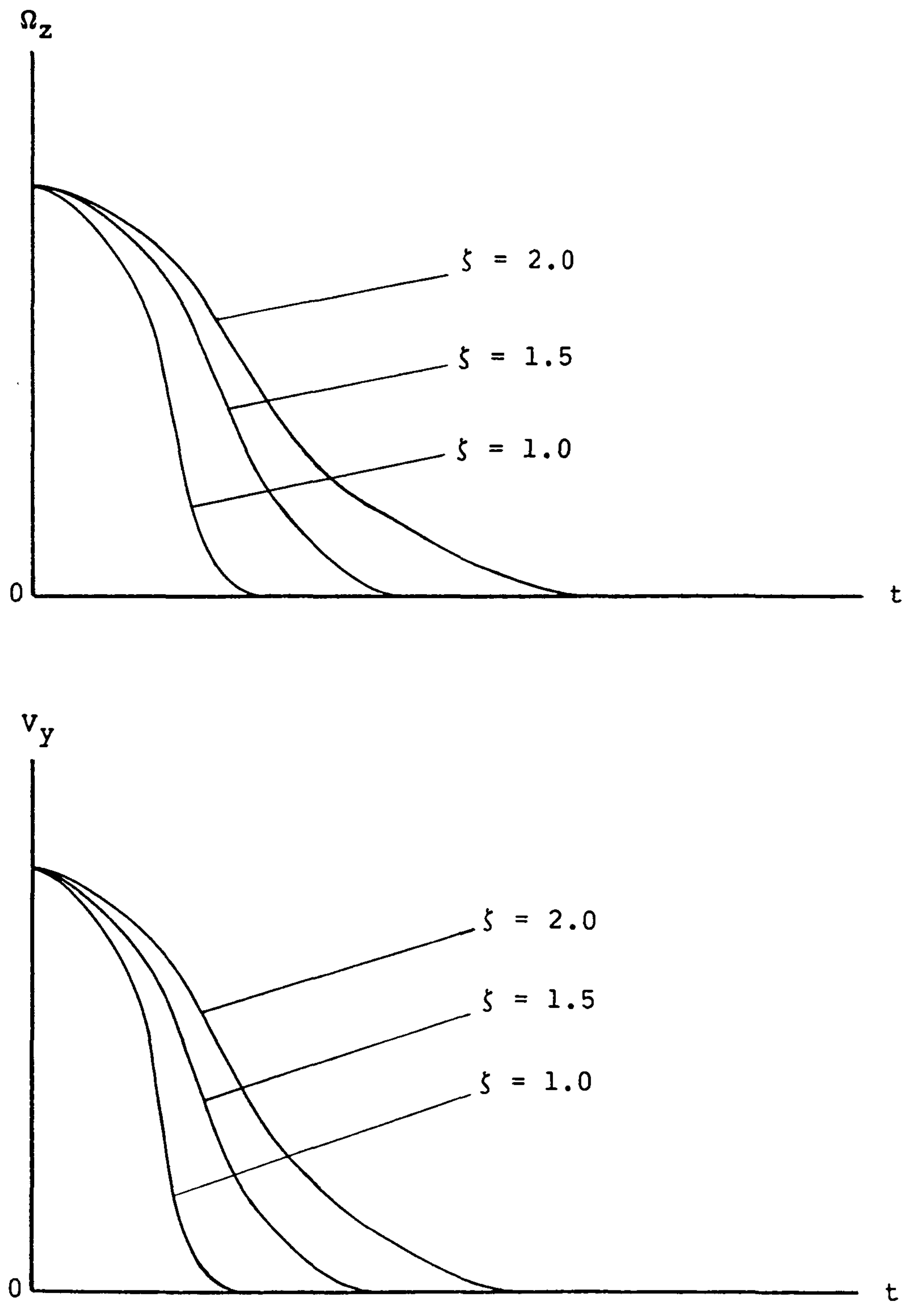


FIGURE 2. Impulse responses of the tractor

system. In a theoretical sense, a relatively larger ζ is better for the system to resist the influence of the external disturbances, but too large a ζ value makes it difficult for the driver to control the tractor in a maneuver or to make an adjustment once the tractor deviates from the preset path.

To date, all the dynamic equations that govern the lateral motion of the system are derived in xoy coordinate system, such as lateral velocity V_y and yaw angle Ω_z. In practice, the tractor operating in field usually travels along a prescribed path with limited width. When subjected to the external disturbances, the tractor will no longer remain on the original path. It is apparent that the state variables Y and yaw angle ψ in the inertia reference frame XOY are important in determining the directional stability of the system.

As described in Fig. 1, by introducing the inertia coordinate system XOY, the absolute motion of the system on the hard, smooth and flat ground can be studied with the aid of rigid body dynamics.

If both the value and the variation of the yaw angle $\psi(t)$ are small, by deleting high order terms, the following approximation holds:

$$\dot{Y} = V_y + V_x \psi \tag{24}$$

where, Y = Lateral displacement of the C.G. of the tractor in the inertia coordinate system XOY.

It is also obvious that the first derivative of the yaw angle is equal to the yaw velocity, that is,

$$\dot{\psi} = \Omega_z \tag{25}$$

where, ψ = Yaw angle of the tractor with respect to X axis in the inertia coordinate system XOY.

The Eqs. (24) and (25) play a role of a bridge that links up the relative motion in xoy plane and the absolute motion in XOY frame.

By substituting the right sides of the Eqs. (24) and (25) with the Eq.(22) and (23), we obtain

$$\dot{Y}=V_x\psi+A((P_1-a)e^{-at}-(P_1-b)e^{-bt})/(b-a) \tag{26}$$

$$\dot{\psi}=B((P_2-a)e^{-at}-(P_2-b)e^{-bt})/(b-a) \tag{27}$$

By taking the infinite integration of the Eq. (27), the general equation of the yaw angle is defined as following:

$$\psi(t)=C_1+B(((P_2-b)e^{-bt}/b)-((P_2-a)e^{-at}/a))/(b-a)$$

If the assumption of the zero-initial condition of $\psi(t)$ is made, the exact solution of the yaw angle $\psi(t)$ can easily be obtained.

$$\psi(t)=\{B[((P_2-b)e^{-bt}/b)-(P_2-a)e^{-at}/a]/(b-a)\}+BP_2/ab \tag{28}$$

Similarly, the general solution of the lateral motion of the system is developed below:

$$Y(t)=r_1e^{-at}+r_2e^{-bt}+r_3t+C_2$$

Given $Y(0) = 0$, the exact solution is obtained.

$$Y(t)=r_1e^{-at}+r_2e^{-bt}+r_3t+r_4 \tag{29}$$

where, $r_1 = (B(P_2-a)V_x/a^2(b-a))-A(P_1-a)/a(b-a)$,

$r_2 = (A(P_1-b)/b(b-a))-B(P_2-b)V_x/b^2(b-a)$,

$r_3 = BP_2V_x/ab$,

$r_4 = -r_1-r_2$.

The corresponding diagrams of the Eqs. (28) and (29) are shown in Fig. 3.

The transient behavior of the system is vividly depicted by the exponential terms in the Eqs. (28) and (29). It can be seen that, when the system is subjected to the impulse disturbances F and M, in the absence of the driver, the system will no longer follow the prescribed course. Instead, it will move along a curved line determined by the Eqs. (28) and (29). After a certain period of time, no more rotation occurs for the system and the tractor will run straight forward with angle ψ_1 from the original path. Therefore, the system is under the condition of directional instability.

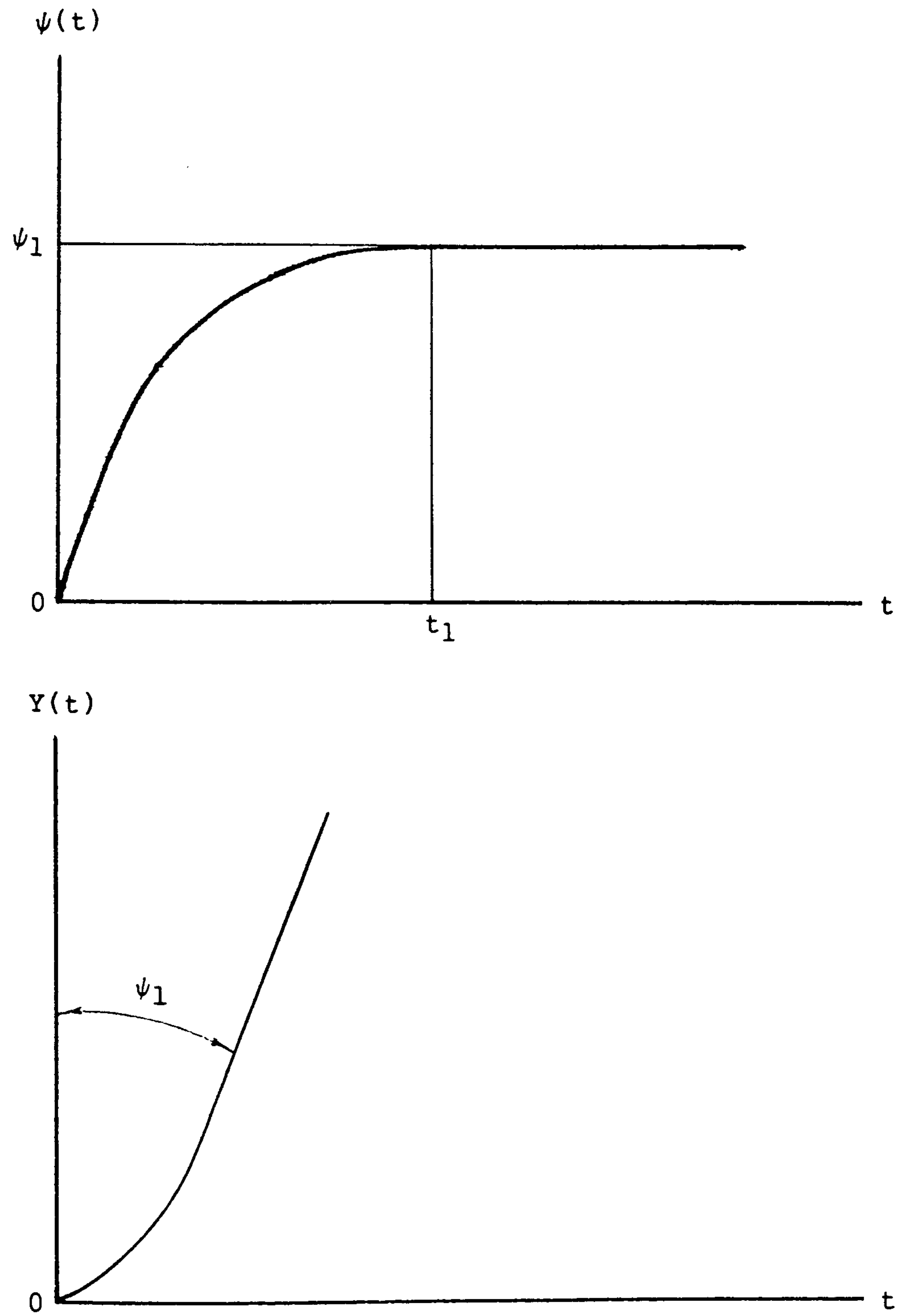


FIGURE 3. Lateral position of the tractor

SIMULATION OF THE DRIVER/TRACTOR MODEL

The results of the previous section has shown that the driver is absolutely necessary in controlling the tractor on desired path and adjusting it against external disturbances. Therefore, the study of the driver/tractor system is of importance in understanding the dynamic behavior of the driver and the interrelationship between the driver and the tractor. In practice, it is difficult to obtain the complicated dynamic behavior of the driver theoretically due to the physiological and psychological reaction of the human body. So the model of the driver as a proportional controller with unit feedback is used to reflect the basic dynamic characteristics of the driver and his effect on the motion of the tractor.

In general, the driver, by movement of the steering wheel, controls yaw angle directly in order to maintain correct orientation and position relative to the preset course. Consequently, it can be hypothesized that the variable of interest in tractor directional control is primarily the yaw angle.

Based on the classical bicycle model of the tractor developed before, by adding the proportional controller with control gain K_p and unit feedback loop, the block diagram representing the driver/tractor closed-loop system is proposed (Fig. 4).

Further assumptions are made that the external disturbing moment $M(t) = 0$, $F(t)$ is a negative vector, and the steering angle δ_f is varying according to the command of the driver. By applying the

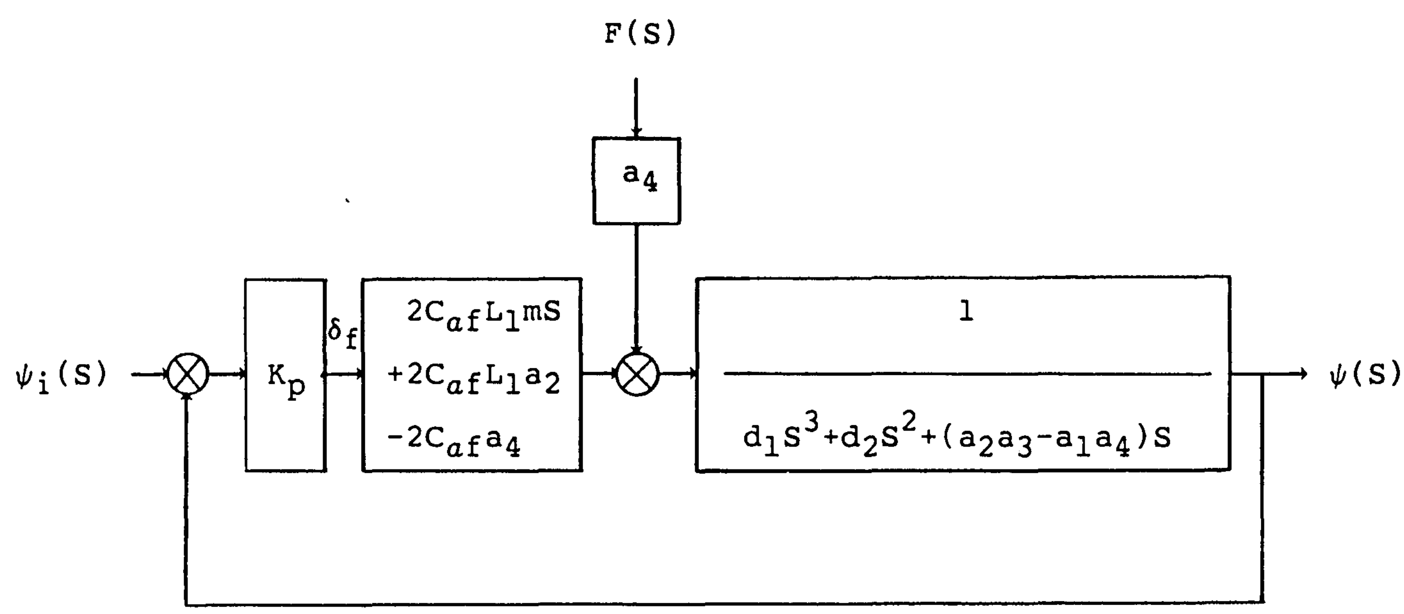


FIGURE 4. Block diagram of the driver/tractor system

Mason's Rule (Palm, 1983) on the driver/tractor system, the corresponding closed-loop transfer function of output yaw angle $\psi(S)$ with the input disturbing force F(S) is presented in Eq. (30).

$$\psi(S)/F(S)=-a_4/(d_1S^3+d_2S^2+d_3S+d_4K_p) \tag{30}$$

where, $d_1 = mI_z$,

$d_2 = a_2I_z+a_3m$,

$d_3 = a_2a_3-a_1a_4+2C_{\alpha f}L_1mK_p$,

$d_4 = 2a_2C_{\alpha f}L_1-2a_4C_{\alpha f}$.

Because of the existence of the internal dynamic properties of the tractor and the external disturbing force, the selection of the proportional control gain K_p appears to be important and is the main problem to be resolved in order to avoid undesirable oscillation of the system in the transient-stage and to reduce the offset error in the steady-state.

Lateral Motion Behavior of the Simulated Driver/Tractor System

A set of data describing the basic characteristics and the operational parameters of a John Deere 4020 tractor is listed in Table 1 (Xie, 1984). Computer simulation was used to obtain the lateral response of this tractor when exposed to a step input disturbing force F(S).

The specific transfer function that describes the dynamics of the driver/John-Deere-4020 tractor system is developed by substituting those data in Table 1 into the Eq. (30):

TABLE 1. Characteristics of a John-Deere 4020 tractor

Tractor geometry (m):	
wheel base	$L = L_1 + L_2 = 2.546$ $L_1 = 1.693$ $L_2 = 0.853$
front wheel diameter	$D_f = 0.47$
rear wheel diameter	$D_r = 1.65$
Tractor mass (kg):	$m = 4000.0$
Tractor moment of inertia ($kg\text{-}m^2$):	$I_z = 5600.0$
Tractor working speed (m/s):	$V_x = 4.0$
Front tire cornering stiffness (kN/rad):	$C_{\alpha f} = 45.0$
Rear tire cornering stiffness (kN/rad):	$C_{\alpha r} = 60.0$
Understeer coefficient (rad):	$K_{us} = -0.073$
Critical speed (km/h):	$V_{crit} = 67.23$

$$\psi(S)/F(S)=-12/(22S^3+640S^2+(4200+610K_p)S+6870K_p) \quad (31)$$

The intent of this simulation is to describe the time response of the driver/tractor system in terms of yaw angle $\psi(t)$ against an external step disturbing force $F(t) = -1000$ N.

As mentioned before, the system is travelling along a straight line on a flat surface, when it is subjected to the lateral step disturbing force $F(t)$ acting at the C.G. of the system. If $K_p = 0$, that is, no a driver in the system, the yaw angle $\psi(t)$ will increase with time and the system becomes unstable (Fig. 5).

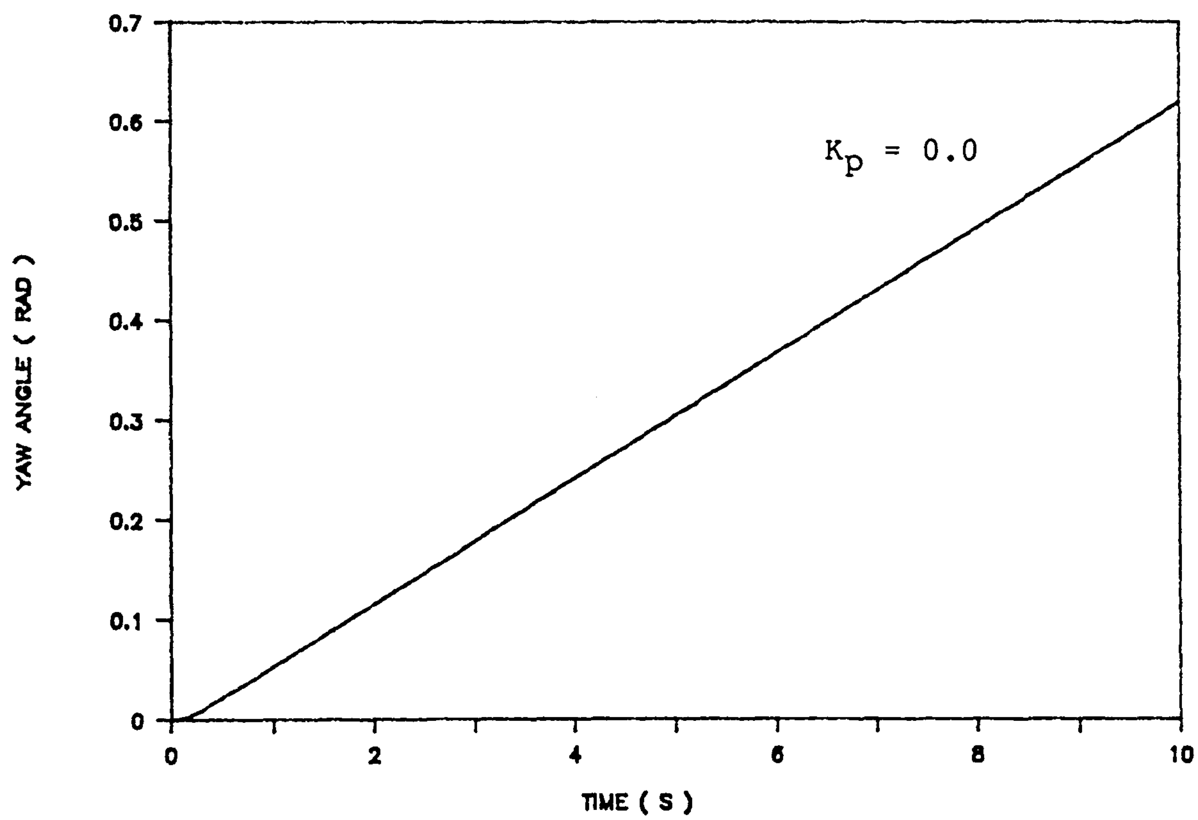


FIGURE 5. Step response of the tractor without control

A small proportional gain K_p, such as $K_p = 10$, is chosen to simulate the driver's effect on the tractor's response $\psi(t)$. The related diagram of the yaw angle of the system is drawn by using LOTUS 1-2-3 and is shown in Fig. 6. It can be seen that the proportional controller responds fast to the transient error and within about 1.0 second the system reaches the steady-state. It is also interesting to note that a slight overshoot occurs at time t = 0.36 s and the oscillation happens in the transient state due to the selection of relatively small control gain K_p. In the steady-state the yaw angle does not approach zero, which leads to the off-set error of the system response.

With the increase of the gain K_p, the overshoot is gradually eliminated and the off-set error is quickly lowered, but the transient oscillation is increased (Fig. 7).

It is also noted that the off-set error always exists (Fig. 8). The final value theorem states that the final value of the time response $\psi(t)$ under the proportional control is:

$$\psi = \lim_{S \to 0} [\ S\ \psi(S)\ ] = 1/6870K_p \tag{32}$$

The Eq. (32) indicates that the off-set error will reduced with the increase of K_p, but the off-set error will never die out no matter what large value of K_p is selected. In practice, the too large K_p can be difficult to implement physically and it also costs too much to reduce the deviation of the response from the set point.

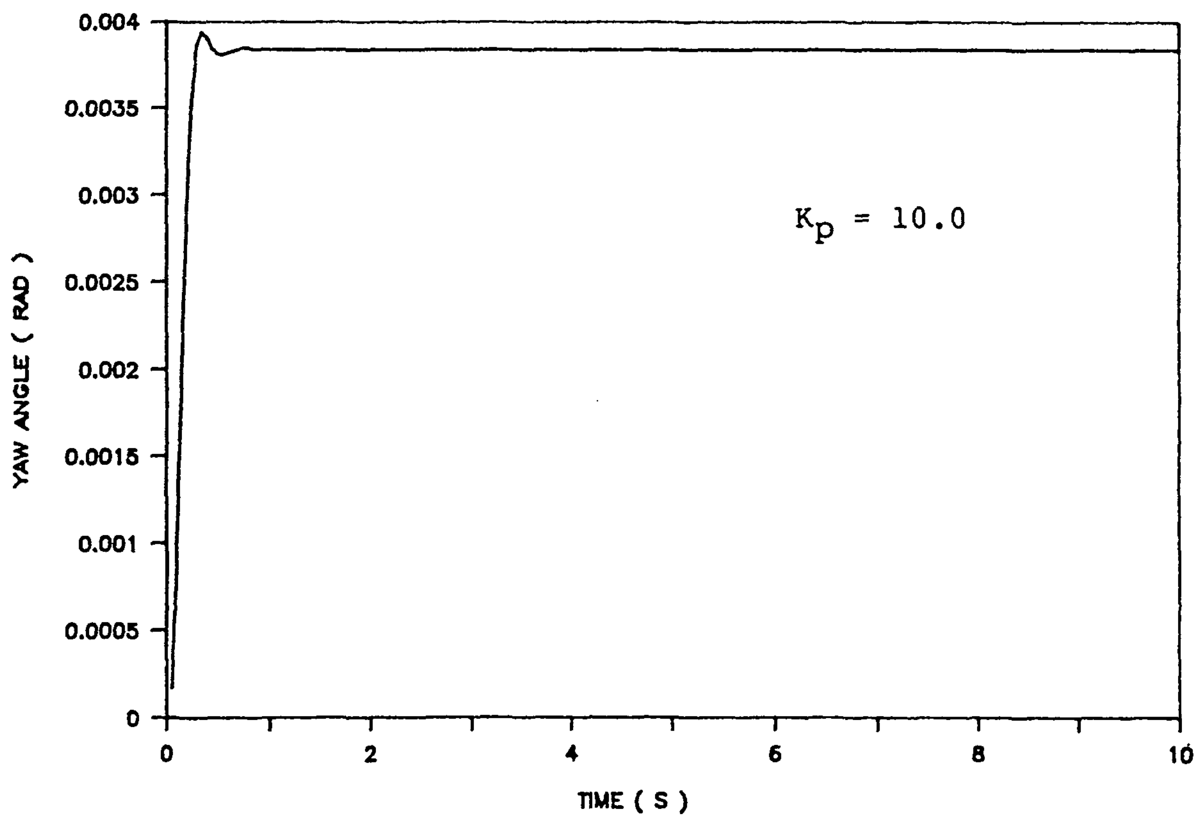


FIGURE 6. Step response with small gain K_p

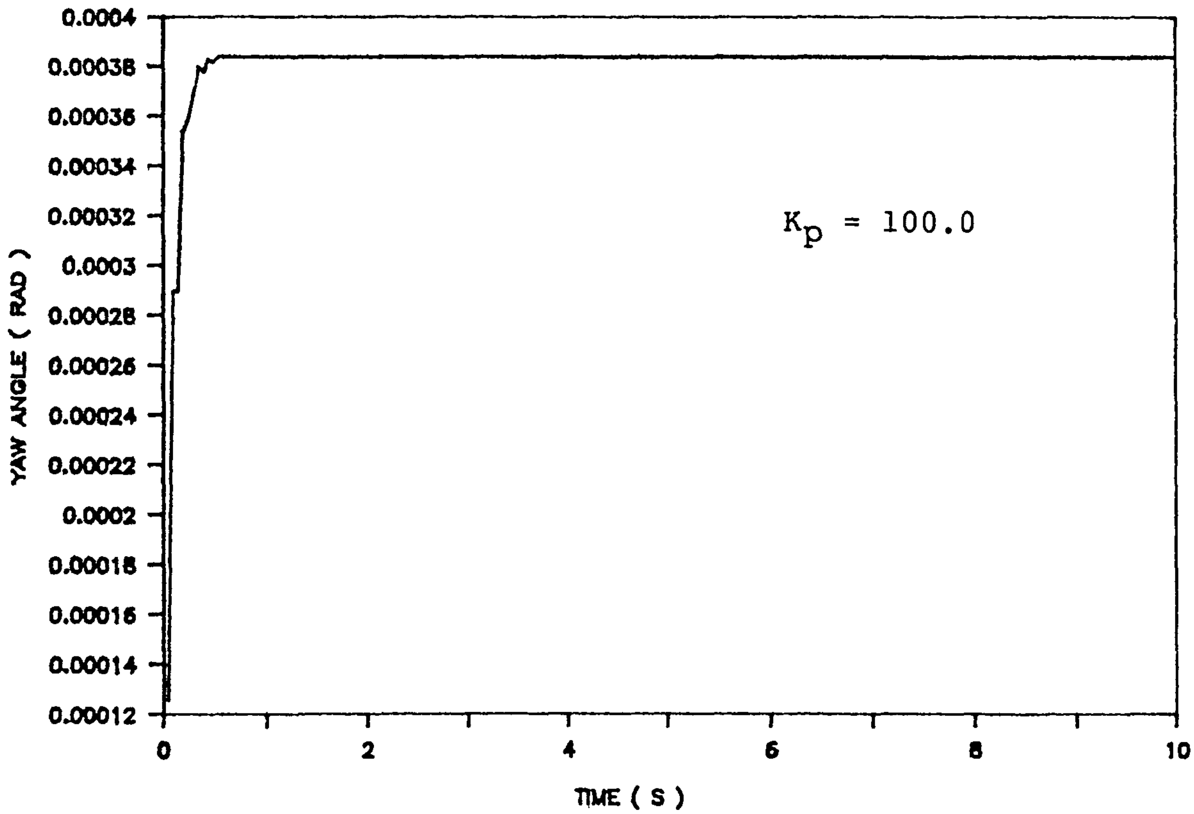


FIGURE 7. Step response with gain $K_p = 100$

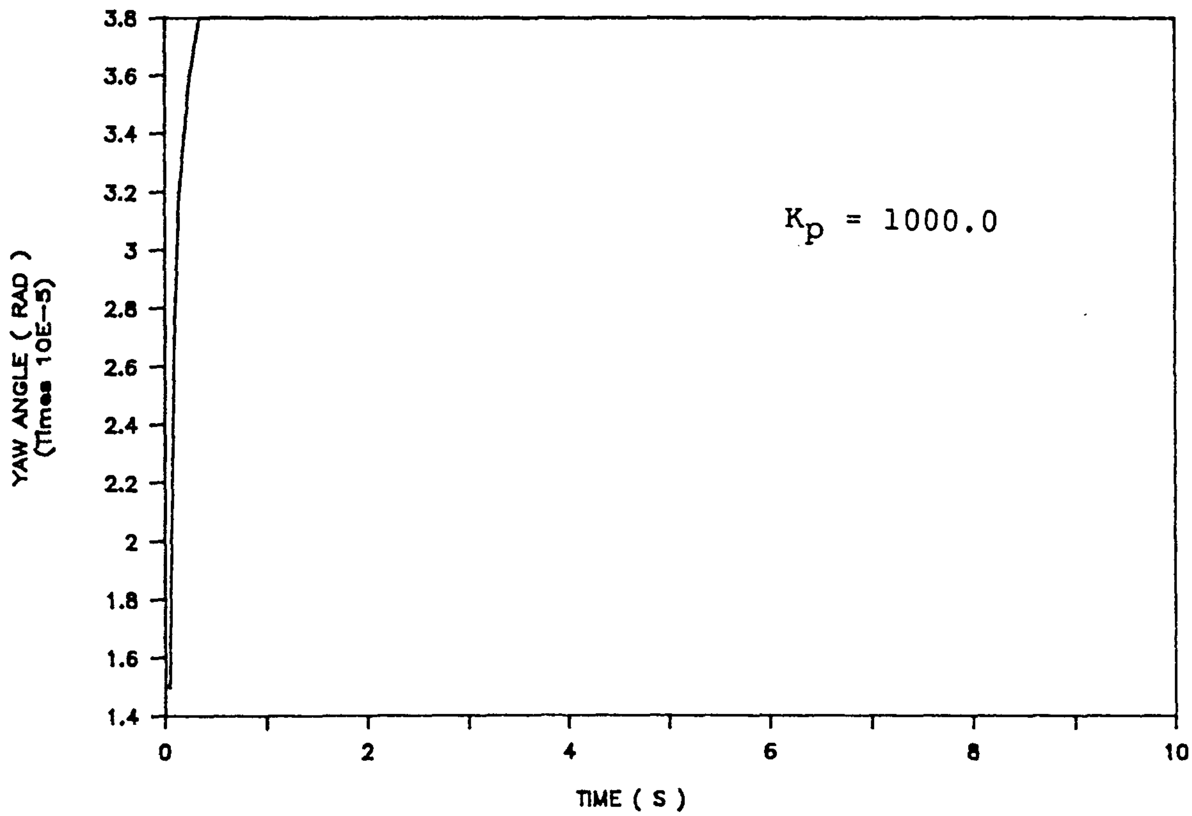


FIGURE 8. Step response with large gain K_p

The simulation has shown that the proportional controller is unable to adjust the system precisely to follow the straight line against the external disturbing force $F(t)$.

SIMULATION OF THE PID-CONTROLLER/TRACTOR SYSTEM

Because of the steady-state error disadvantage of the proportional controller, a three-mode conventional PID (Proportional-Integral-Derivative) automatic controller is introduced in this section. The off-set error might be eliminated through the introduction of the integral control with the control gain K_i, in which mode the the control signal increases as long as the error is nonzero. There is the potential for instability caused by the continuous error integration even after the disappearance of the off-set error. Derivative control with the control gain K_d has been added to detect whether error is still present. Therefore, the three different control modes complement each other and form the most popular combination of the automatic controller.

The computer programs are presented in Appendix A. Computer graphic techniques are employed to describe the yaw angle of the PID-controller/tractor system in time-history for various simulation runs. In the simulation, different sets of control gains are selected to investigate the directional stability of the system under both step and random disturbing inputs.

Performance index ITSE (Integral Time Squared-Error) and other specifications such as percentage overshoot and settling time are evaluated as the criteria in the simulation for the determination of the 'optimal' control gains that 'best' maintain the system on the required course against external lateral disturbances.

Computer Simulation Algorithm

Time response

Both the main program and subprograms are written in standard FORTRAN and were successfully run on an IBM-compatible microcomputer to simulate the yaw angle of a tractor represented as a bicycle model.

The time response of the system was obtained using Laplace-transform techniques. The first step was to factorize the denominator polynomial in $\psi(S) = N(S)/D(S)$ using RPOLY subroutine. RPOLY is a three-stage algorithm for finding the roots of real polynomials. This algorithm has been described by Jenkins and Traub (1970). All factors are retained as binomials of the form $(S+a\pm bj)$. The factors are sorted first by their imaginary parts and then by their real parts by a subroutine CSORT. The subroutine also forces parts that are nearly zero (TOLZ = 1.0E-06) to be exactly zero and CSORT also forces equality among parts that fall within a relative-error criterion (TOL = 1.0E-04). A factor-repetition count subroutine, FCTCNT follows CSORT and yields an integer array indicating the multiplicity of the factor. It is recognized that digital computers can find individual partial fraction coefficients for $(S+a-\epsilon)$ and $(S+a+\epsilon)$ even if $\epsilon \simeq 1.0E-05$, but the subroutines had been developed for use in teaching (AE503) and were used for convenience. The program will find partial fraction coefficients for both distinct and repeated factors. As the final step in the time response calculations, terms of the form $C_i e^{-r_i t}$ for $C_i/(S+r_i)$ or $C_j t^m e^{-r_j t}/m!$ for $C_j/(S+r_j)^m$, are calculated at each time step for each factor. Complex arithmetic is used and the response at any time step is the real part of the sum of all the complex response components. Although the imaginary part of the total time response should be zero, because complex factors always appear as conjugates, it was observed that there was a small residual imaginary sum. This sum was typically less than 10^{-5} x 'the real sum'.

Optimization

The performance index chosen was the Integral Time Squared Error ITSE (Palm, 1983). This was evaluated using Simpson's rule applied to the time response results. The time response program and the ITSE program are combined into a function having arguments the three controller gains K_p, K_i, and K_d. 'Optimum' gains were determined with a multi variable unconstrained minimization routine translated from pseudo code in Dennis and Schnabel (1983). In brief, the program FCNMIN estimates the next step that should be taken from an estimate of the minimum by applying the Newton technique. The fraction of the Newton step that is taken for each iteration is determined by a line search algorithm that models the function with a quadratic or cubic over the Newton step. When the minimum is being approached closely, this model allows the algorithm to estimate the minimizing values of the independent variables quite rapidly. The Hessian that is used to calculate the Newton step is estimated with a secant update and this update is very computationally economical. The computation algorithm for the time response and its optimization is listed using a flow-chart in Fig. 9.

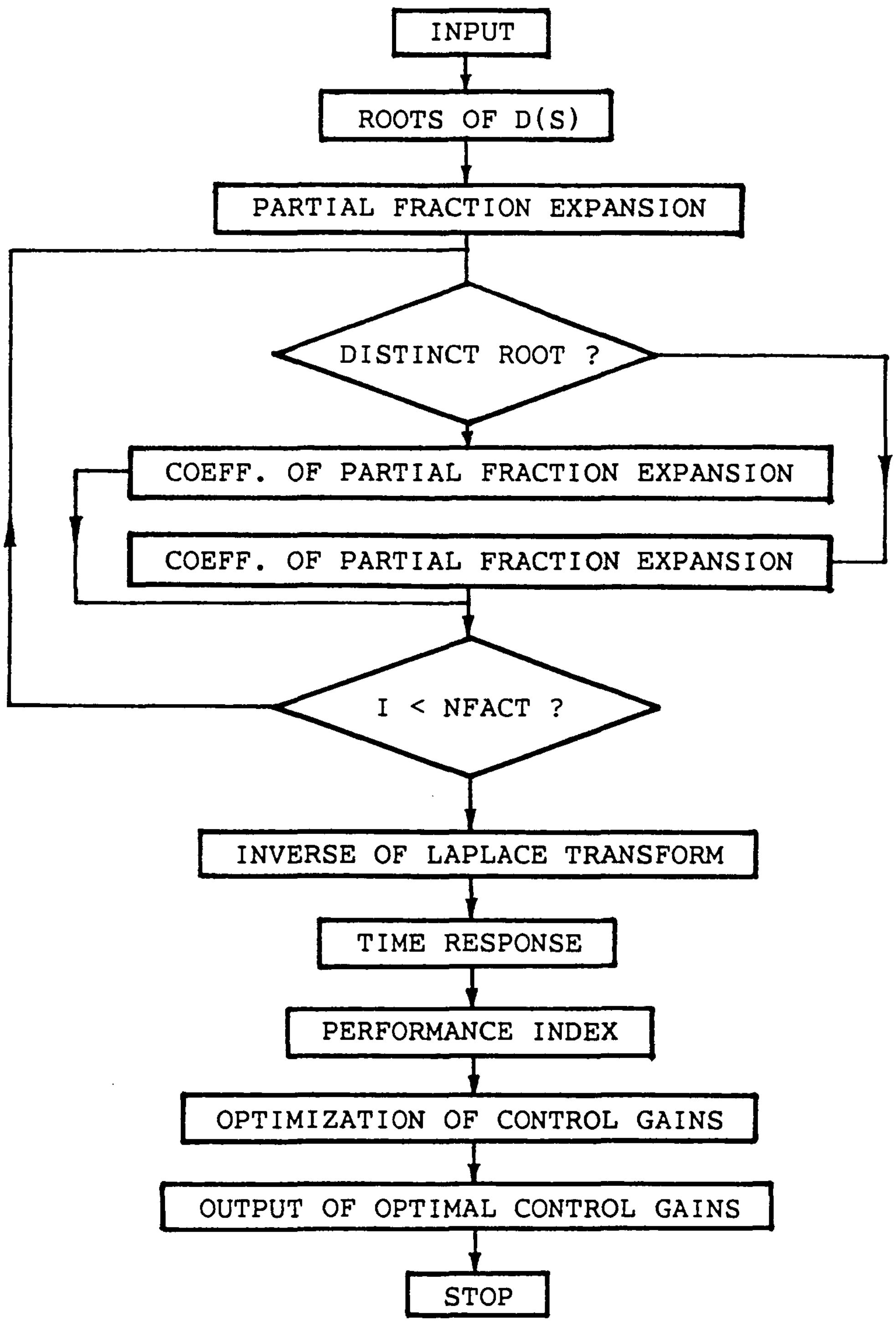


FIGURE 9. Computer simulation algorithm

Simulation of Step Response of the PID-Controller/Tractor System

Based upon the bicycle model of the tractor, with a PID controller, the block diagram of the PID-controller/tractor system is drawn as in Fig. 10.

In the light of Mason's Rule (Palm, 1983), the transfer function of the yaw angle $\psi(S)$ versus a step disturbing input F(S) is developed in Eq. (33).

$$\psi(S)/F(S)=-12S/(b_1S^4+b_2S^3+b_3S^2+b_4S+b_5) \tag{33}$$

where, $b_1 = 22$,

$b_2 = 640+610K_d$,

$b_3 = 4200+610K_p+6870K_d$,

$b_4 = 610K_i+6870K_p$,

$b_5 = 6870K_i$.

The simulation of the change of the yaw angle of the automatically controlled tractor, which moves at 4 m/s straight ahead on a flat and hard plane against the step disturbing force $F(t) = -1000$ N, will be based on the Eq. (33) and the results of the step response for different sets of control gains will be explained in detail next.

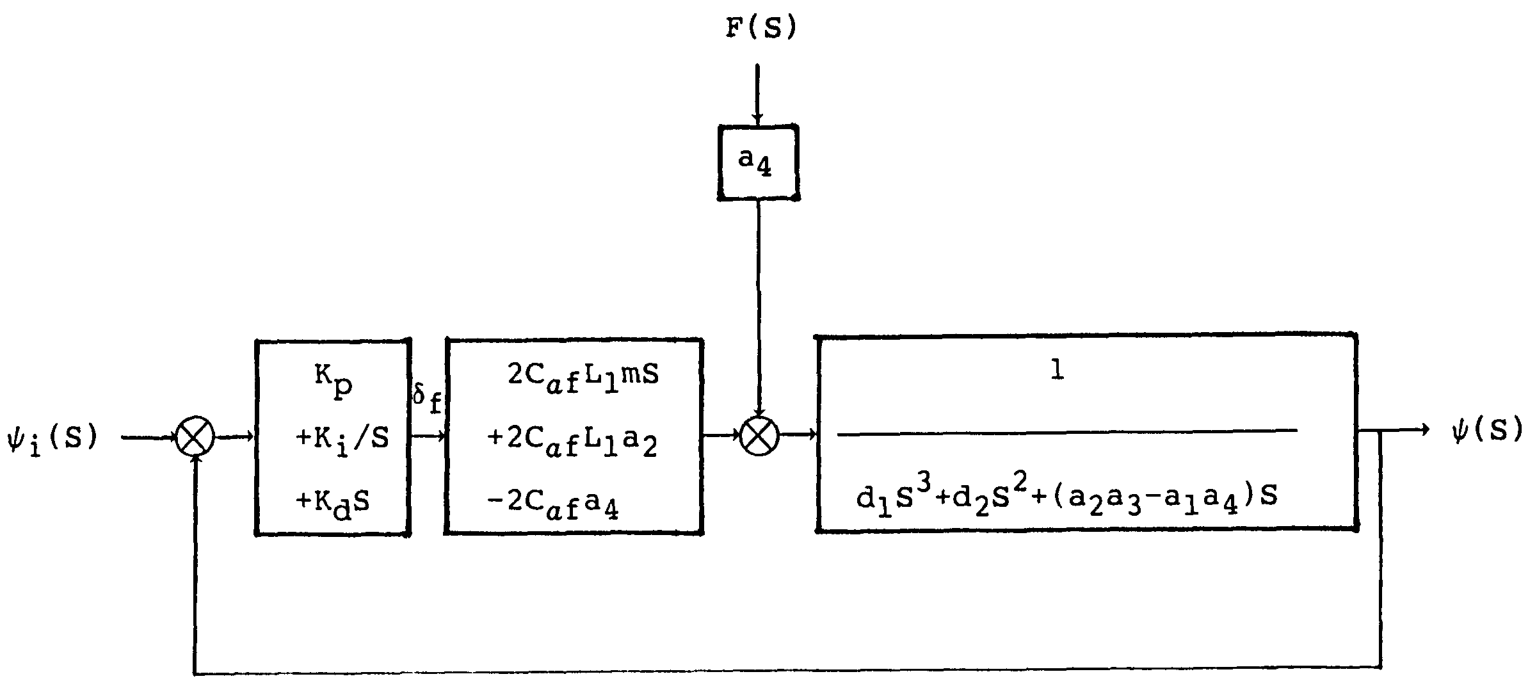


FIGURE 10. Block diagram of PID controller/tractor system

Simulation 1: The effect of K_p on step response

$K_p = K_i = K_d = 1$

For the first simulation run, the initial values for those three control gains are chosen as 1 for the simplicity. Fig. 11 has shown the step response, by the yaw angle $\psi(t)$, within the time periods of 10 seconds. It can be seen that the amplitude of the response curve are reduced with time, but the curve oscillates too much with large overshoot (18.5 rad) and undershoot (-5 rad) at t = 1.7 s and t = 6.0 s, respectively.

Routh-Hurwitz criterion states that the necessary and sufficient condition for a stable system is that the first column of the Routh array contains no sign change. The Routh array derived from the characteristic equation of the closed-loop system (see Eq. (33)) is:

22	11680	6870
1250	7480	0
89	6870	
-89000	0	
6870		

It is obvious that the system is unstable theoretically due to the existence of one negative value in the first column of the Routh array.

The main reason for the oscillation, overshoot and undershoot is due to the insufficient control gains of the controller that corrects the response error. The simulation has used the ITSE as the performance index that reflects the response error in the given time period and measures the quality of the transient response of the system. By checking ITSE = $\int_0^{10} te^2(t)dt$ = 1647.6, it is obvious that the system is out of control in the transient state.

K_p = 100, K_i = K_d = 1

When the proportional gain K_p increases from 1 to 100 while K_i and K_d remain the same, the oscillation of the step response is greatly reduced and both overshoot and undershoot are eliminated (Fig. 12) due to the larger K_p that effectively detects and adjusts the deviation of the yaw angle from the set point ($\psi_i(t) = 0$). When the system is subjected to the step disturbing force F(t), the yaw angle at first increases quickly and reaches the positive maximum ($\psi(t)$ = 0.382 rad) at time t = 0.65 s. Then, $\psi(t)$ decreases slowly with time. Since the integral gain K_i is small, the controller is unable to sufficiently feed back an increasing error signal in respect to the continuous deviation of the system, which results in the slow reduction rate of the error.

k_p = 1000, K_i = K_d = 1

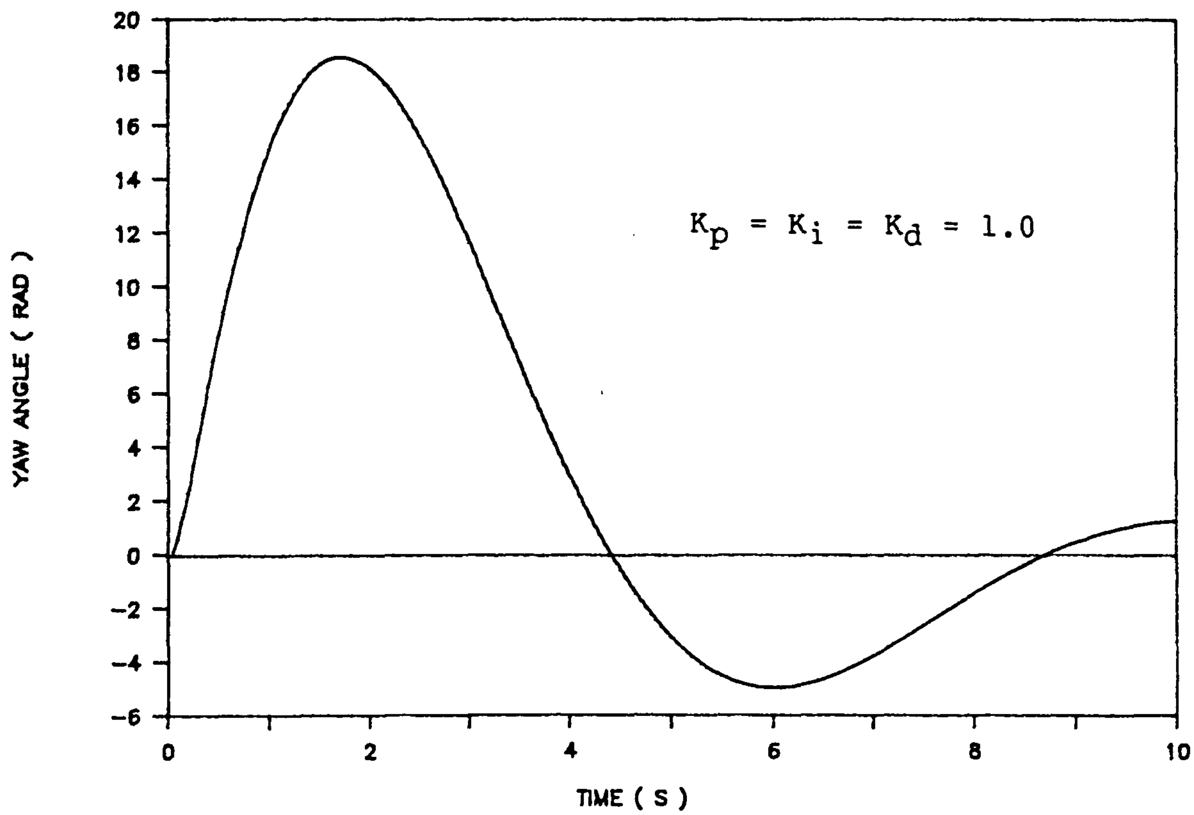


FIGURE 11. The effect of $K_p=K_i=K_d=1$ on step response

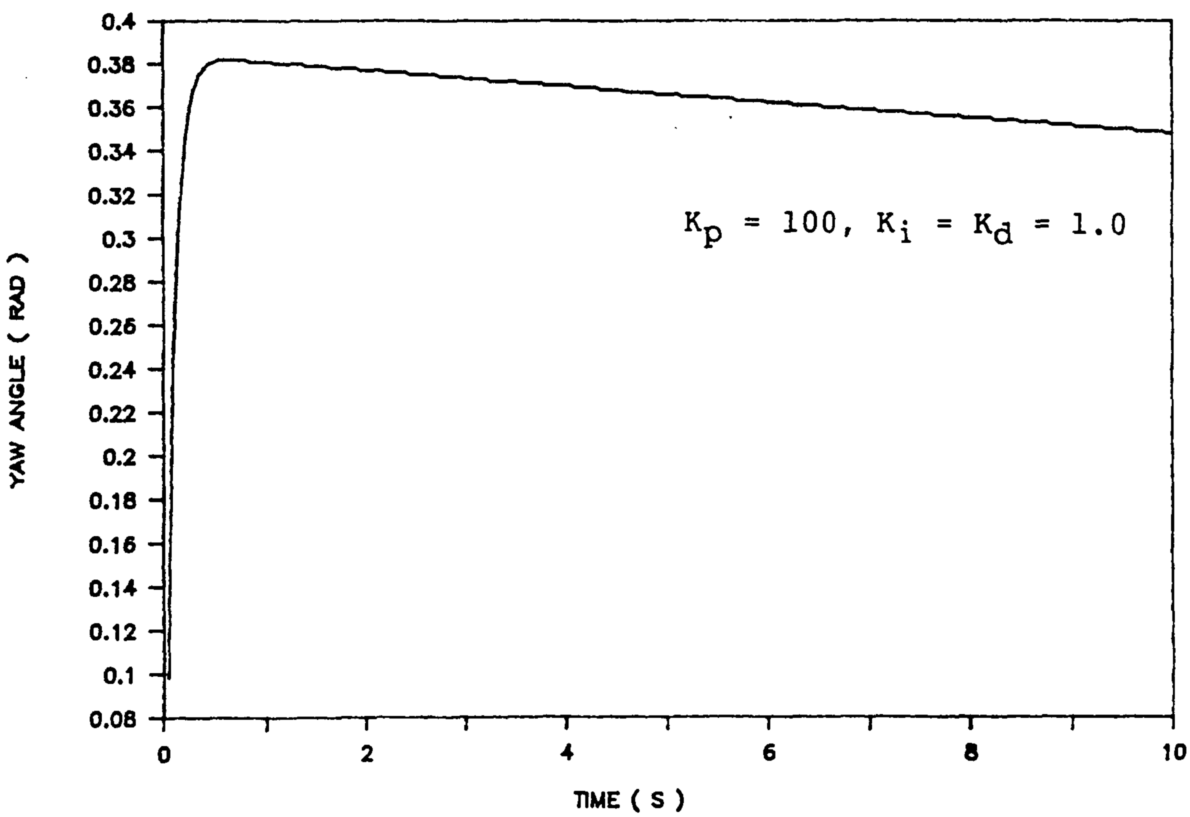


FIGURE 12. The effect of the change of K_p on step response

With further increase of K_p, the step response becomes even better (Fig. 13). It can be seen that the deviation of the response is small (ITSE = 0.0729), which indicates the 'goodness' of the transient response in terms of yaw angle $\psi(t)$. The output curve is similar to that in Fig. 7 except that this curve is nearly horizontal over the given time period. It should be noted that, because of the small values of both integral gain K_i and derivative gain K_d, the function of this PID automatic controller is actually equivalent to that of the proportional controller mentioned previously. Therefore, no matter how large K_p is, theoretically, the off-set error is always exists in the output.

Simulation 2: The effect of K_i on step response

$K_i = 100$, $K_p = K_d = 1$

If K_p and K_d are fixed to 1 and K_i is increased to 100, the step response of the system will be totally different from that of previous simulations (Fig. 14). The yaw angle becomes a periodic function with its amplitude increasing with time. The curve has shown that the overshoot occurs at time t = 0.35 s with the amplitude 2.5 radian. Due to the increasing oscillation of the curve with high frequency, the motion of the tractor will be accompanied with severe swing and finally be in an unstable situation. This unstable condition can also be

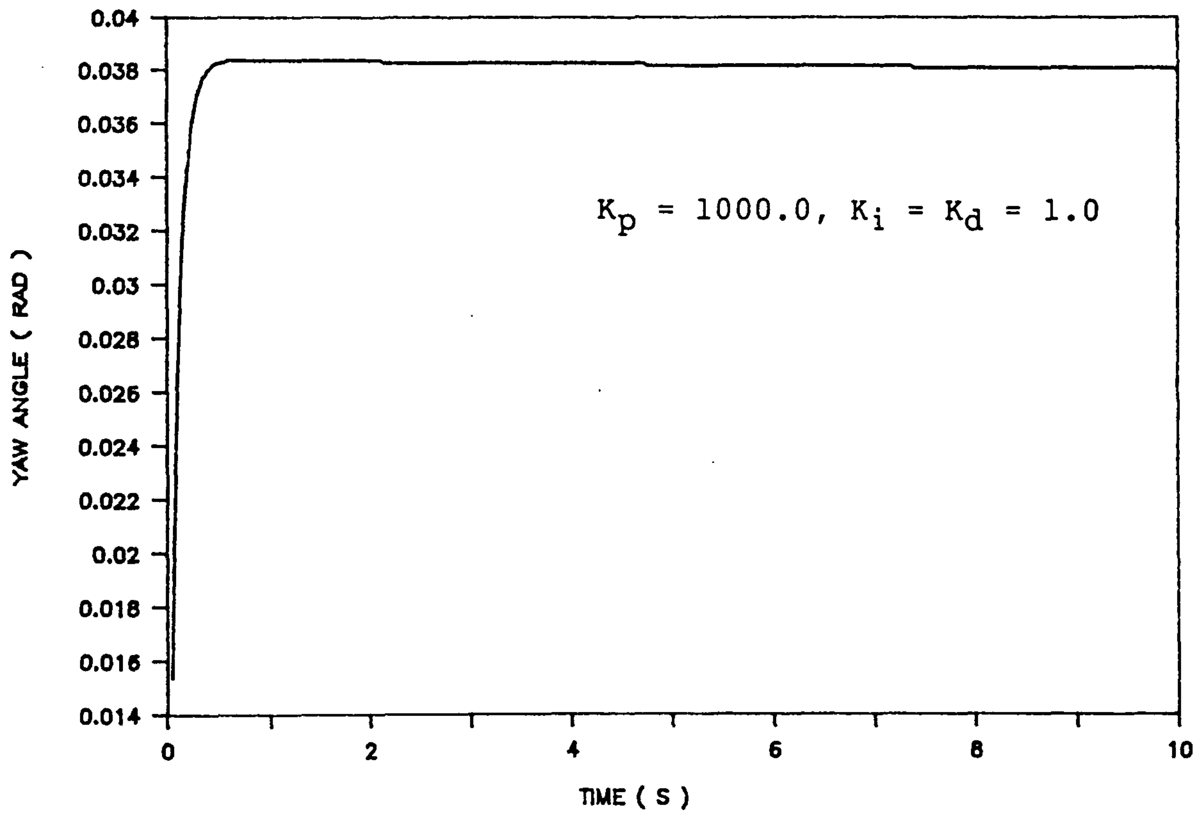


FIGURE 13. The effect of large K_p on step response

proved by the Routh-Hurwitz criterion. The corresponding Routh array of the characteristic equation in Eq. (33) is:

22	11680	687000
1250	67870	0
10485	678700	
-14033	0	
687000		

Since the members in the first column of the Routh array are not all positive, the system is unstable theoretically.

$K_i = 1000$, $K_p = K_d = 1$

In order further to investigate the influence of the integral gain K_i on the yaw angle $\psi(t)$, large K_i is selected for the PID controller. Fig. 15 indicates that the larger the K_i, the worse the step response. It is also interesting to note that the oscillation increases greatly with even higher frequency. The simulation shows that after time t = 5.0 s the yaw angle tends to approach to infinite. It is concluded that the integral function alone is unable to adjust the deviation of the system against the step disturbance but worsens the output.

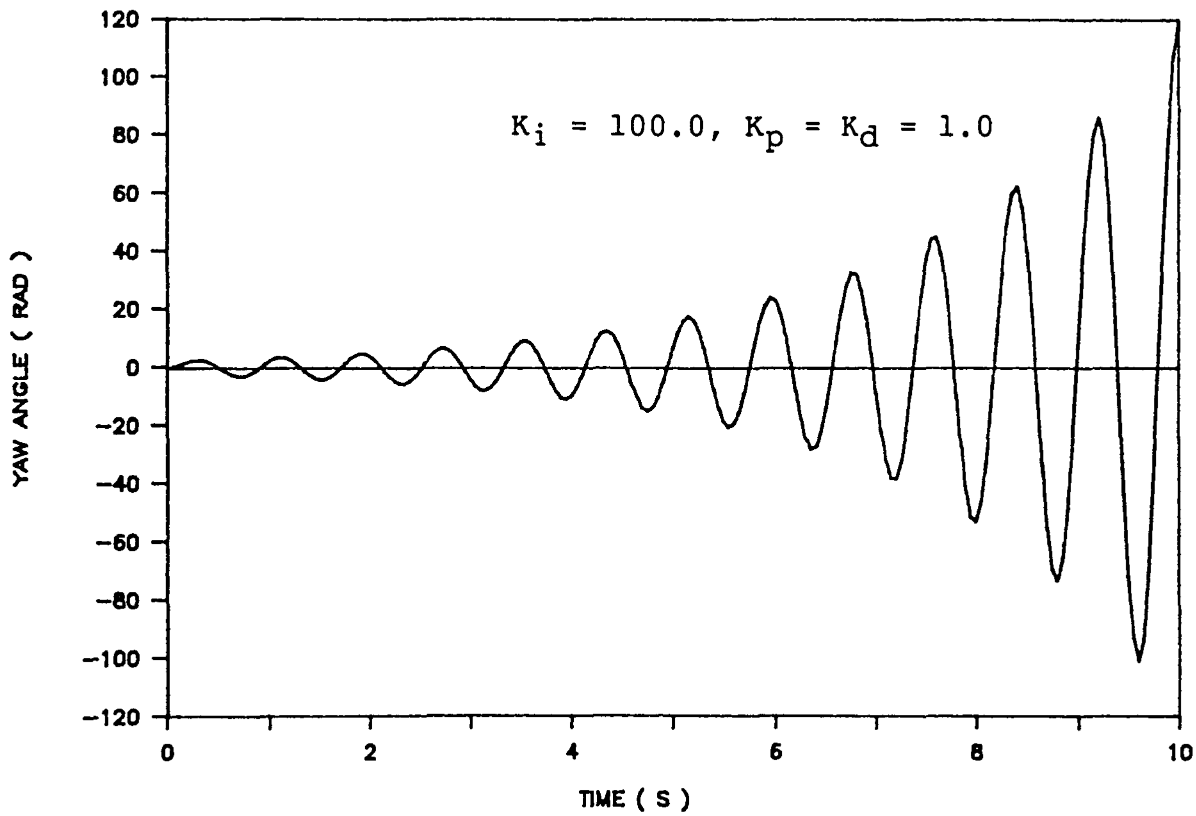


FIGURE 14. The effect of the change of K_i on step response

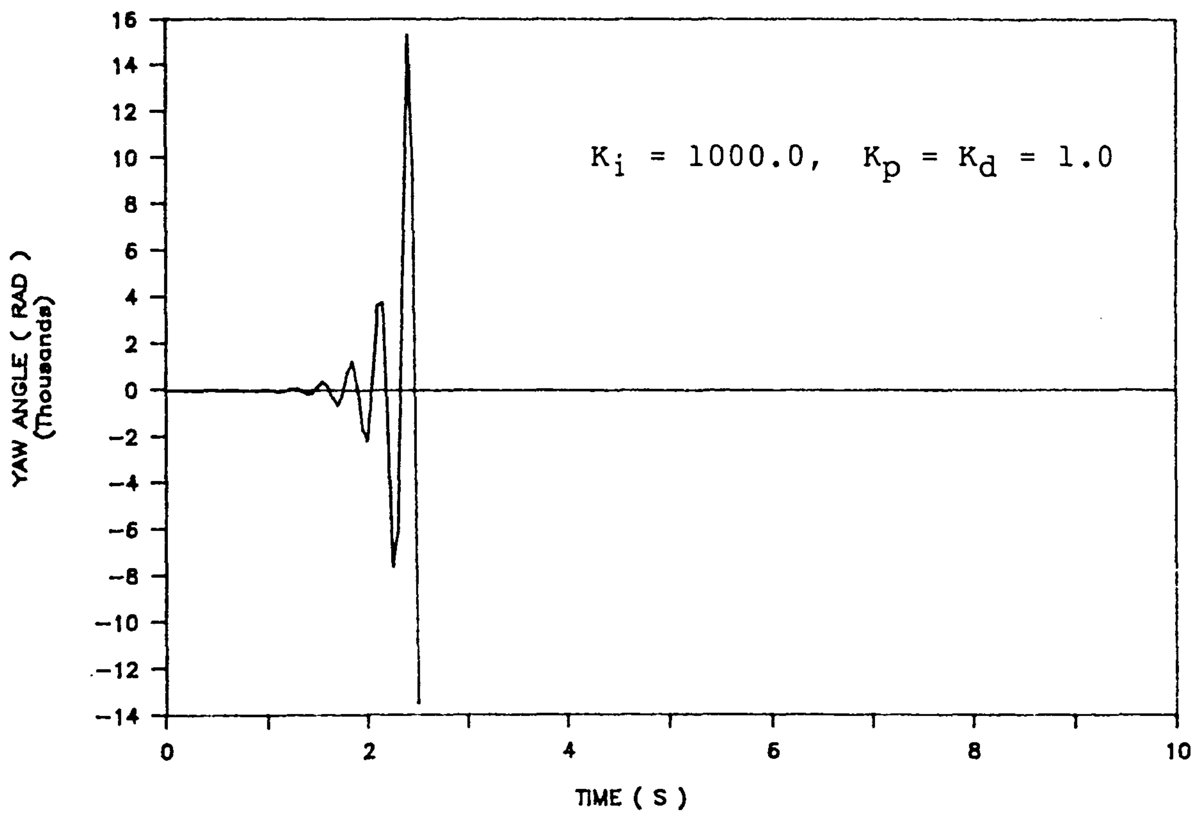


FIGURE 15. The effect of large K_i on step response

Simulation 3: The effect of K_d on step response

The major effect of the derivative gain K_d on the step response of the system is shown in Fig. 16. In general, the yaw angle $\psi(t)$ increases slowly with time, but no initial overshoot and oscillation exists. When the value of 100 is assigned to K_d and the other control gains maintain the same quantity of 1, the deviation rate of the yaw angle is somewhat large and the system becomes unstable ($\psi(10) = 3.05$ rad) within the time periods of 10 seconds.

With the increase of K_d, the deviation of the yaw angle from the prescribed value is reduced considerably (Fig. 17). The simulation indicates that the derivative gain K_d, on some extent, resists the output deviation, but its influence on the system response is so small as not to prevent the system from instability when the time period is extended.

It can be seen from the previous simulations that the proportional gain K_p plays an important role in reducing the response error of the system in the transient state. The function of the integral gain K_i alone is to destabilize the directional motion behavior of the system. The derivative gain K_d by itself does no significant harm to the response, but the use of it as the only major item in the controller is a futile effort for control purpose. It is the proper composition of these three gains that can generate a 'perfect' controller in controlling and adjusting the system against the external disturbances.

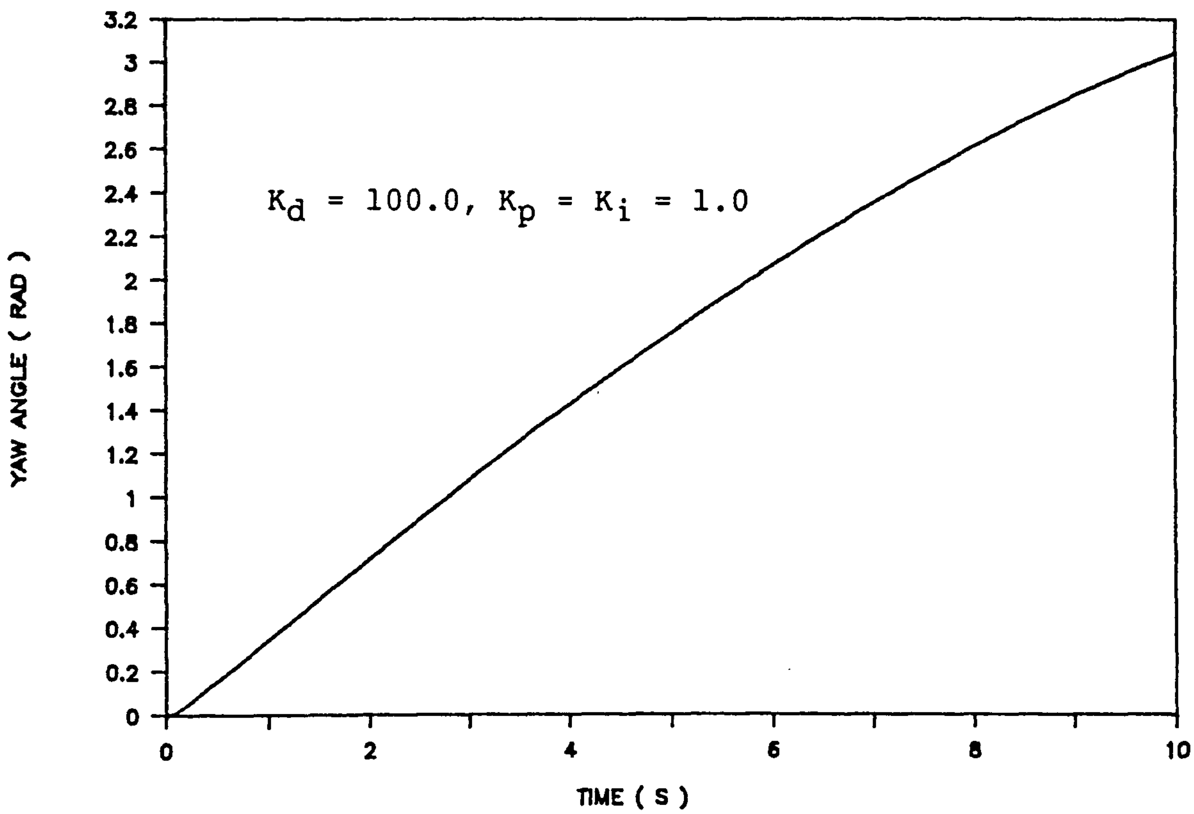


FIGURE 16. The effect of the change of K_d on step response

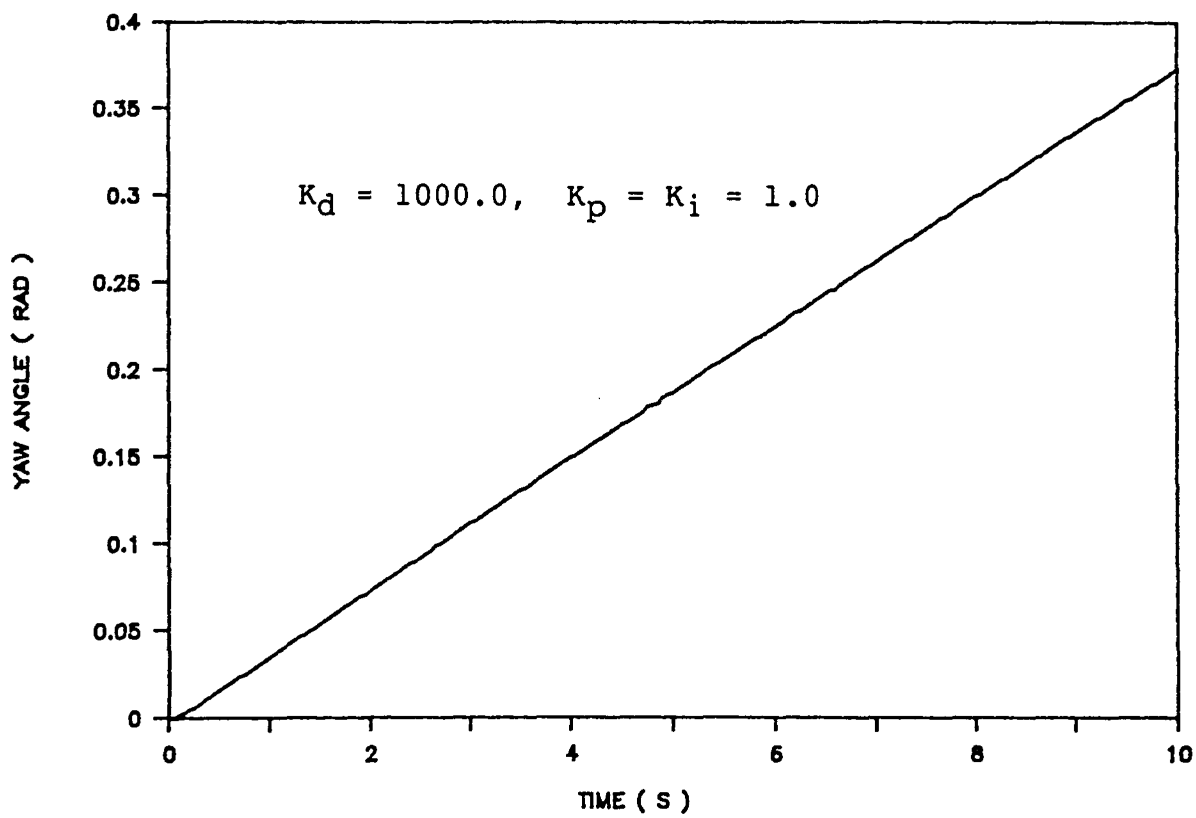


FIGURE 17. The effect of large K_d on step response

Therefore, how to select a 'optimal' PID controller that minimizes the response error, overshoot and oscillation is becoming the main goal in the simulation of the automatic control for the tractor guidance.

Since the system is of fourth-order with three control gains to be selected, it is quite difficult for the gains to satisfy all the performance specifications that are required for an ideal system. So the synthetic performance index ITSE has been introduced to specify the system's desired performance theoretically.

The Trial-and-Error method will be utilized in the following simulation procedure for the selection of a set of 'optimal' control gains. Although the Trial-and-Error method is time consuming, the practically optimal control gains can be easily obtained with the aid of digital computer.

Simulation 4: The effect of K_p and K_i on step response

The previous simulations have shown that the proportional gain K_p has great influence on the deviation control of yaw angle $\psi(t)$. From Fig. 5 it can be seen that, with the increase of K_p, the off-set error becomes smaller and smaller; but it will never vanish no matter how large K_p is. By considering the physical limitation of designing K_p in practice, and if ITSE = 10^{-7} is required, a reasonable value of K_p = 1000 can be chosen. In order to eliminate the off-set error, the integral gain is introduced. Fig. 18 has indicated the simulated results of the step response under the variation of integral gain K_i.

When $K_i = 10$, the deviation of the yaw angle is relatively large (0.038 rad) and its reduction rate of the error is extremely slow due to the small K_i. If K_i is increased to 100, the time response is much improved; but the output curve still indicates that K_i is not large enough to eliminate the off-set error within the given time period of 10 seconds. When K_i is further increased and chosen as 1000, the off-set error reaches zero at t = 7.0 s. It is also interesting to note that the PI action does not cause any kind of oscillation. Although the overshoot appears at large K_i, its value is quite small (0.03 rad). It is also obvious that, with the increase of K_i, the settling time of the system response curve is reduced enormously, which is important to the system that requires an accurate path-following and sensitive characteristics.

It should be mentioned that K_i is likely to generate a control signal even after the error has disappeared, which might lead to the severe oscillation of the system response. One way to detect the error existence and the error rate is to use the derivative action K_d. The next simulation will focus on the study of the step response of the system under the change of derivative gain K_d.

Simulation 5: The effect of K_p, K_i and K_d on step response

Fig. 19 illustrates the changes of the yaw angle $\psi(t)$ resulting from the changes of K_d ($K_p = K_i = 1000.0$). Three different values of K_d are used to investigate its influence on the yaw angle. It is

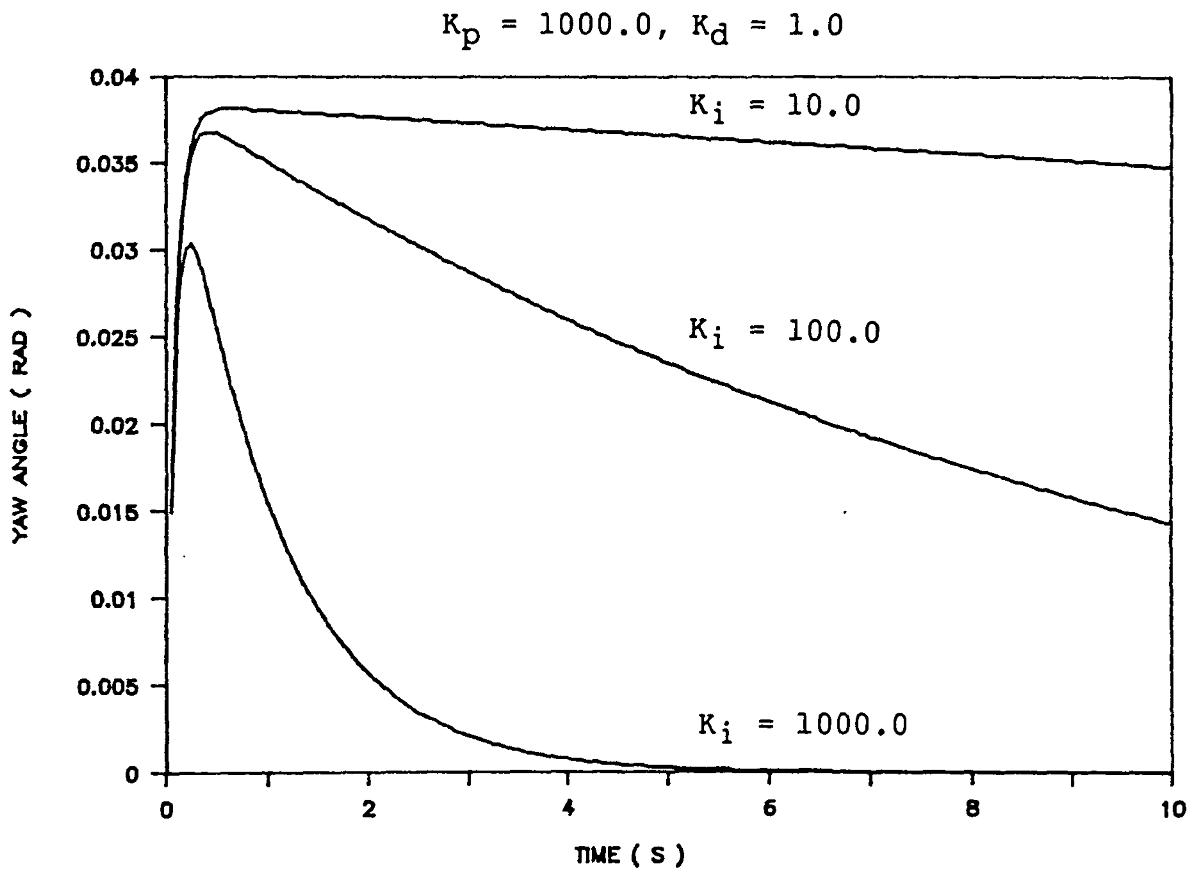


FIGURE 18. The effect of K_p and K_i on step response

apparent that K_d does not affect the step response considerably. Three response curves have the similar shape in spite of the change of K_d. But it should be noted that for the smaller K_d the response error will die out fast. If too large K_d is selected, such as $K_d = 300$, the response oscillation will happen because of the excessively sensitive function of K_d. By the Trial-and-Error method, a 'optimal' PID controller is obtained with $K_p = K_i = 1000$ and $K_d = 100$, which gives a desirable lateral step response of the system.

Optimization of the PID Controller

In the previous section, the Trial-and-Error method was applied to investigate the effect of each control gain on the yaw angle of the tractor under the step disturbing input. Because of the high order of the transfer function and the existence of three control gains, it is extremely difficult to determine exactly the optimum control gains that minimize the performance index ITSE by the use of the Trial-and-Error method. So this section will present a numerical method to resolve this optimization problem. The corresponding optimization routine named FCNMIN.FOR has been written to fulfill this task (Appendix B).

The initial input values of these three gains for the optimization simulation are chosen arbitrarily and after several iterations the three gains are obtained as following:

$K_p = 376.0$

$K_i = 2440.0$

$K_p = K_i = 1000.0$

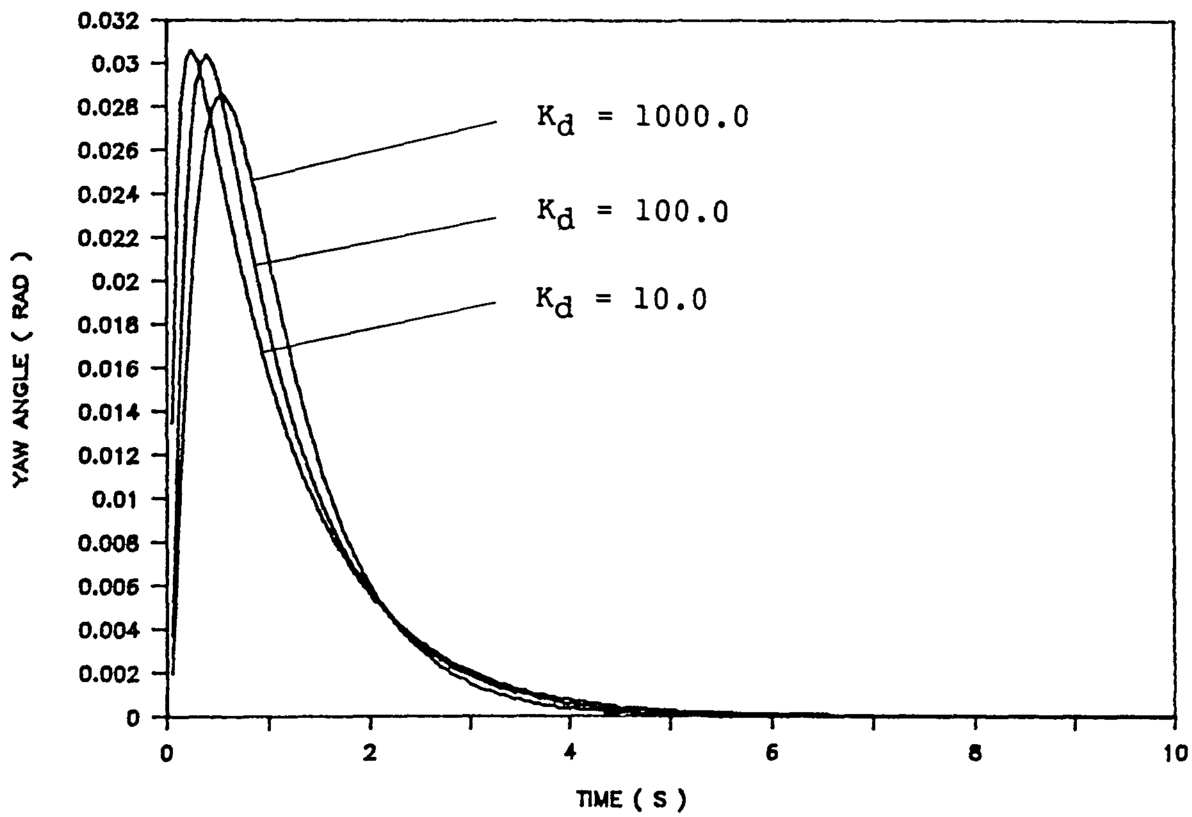


FIGURE 19. The effect of K_p, K_i and K_d on step response

$K_d = 0.756$

The related step response of the tractor is plotted in Fig. 20. It can be seen that, when the lateral disturbing force F(t) = -1000 N is acting at the C.G. of the tractor, the controller with these particular gains is able to correct the response error and effectively pull the tractor back to the prescribed path within one second. The performance index has indicated that the error is small (ITSE = 9.278×10^{-5}). The figure also shows that there exists slight oscillation when the tractor approaches its maximum deviation (0.052 rad).

The next loop of the FCNMIN automatically produce another set of control gains:

$K_p = 415.6$

$K_i = 6750.0$

$K_d = 0.7564$

This time the amplitude of the overshoot becomes even smaller (0.032 rad) and the transient-state length becomes even shorter (0.70 s), but the entire transient stage is accompanied by the increasing oscillation (Fig. 21).

Compared with the gains obtained previously, it is apparent that K_p and K_d rise slightly but the K_i increases greatly. The larger K_i, on one hand, reduces the off-set error, and on the other hand, also causes the system oscillate with higher frequency (Fig. 22).

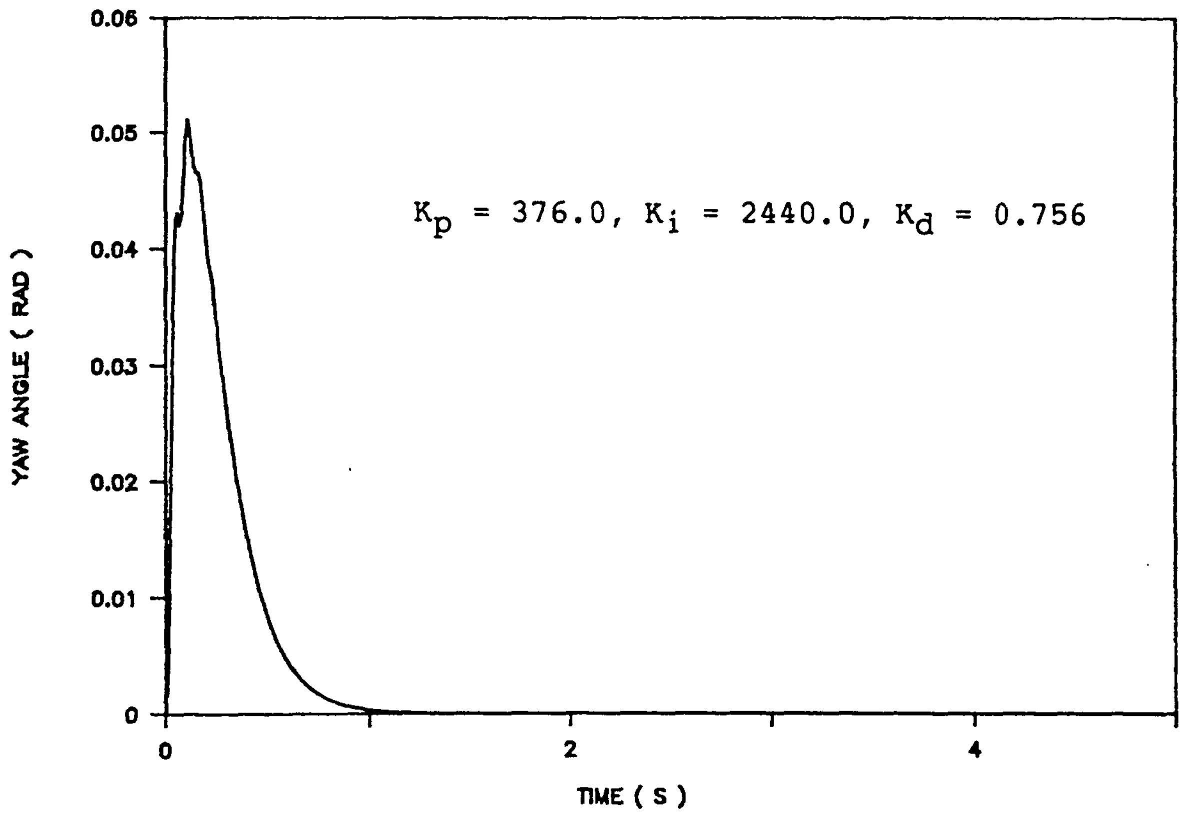


FIGURE 20. Step response of PID controller/tractor system

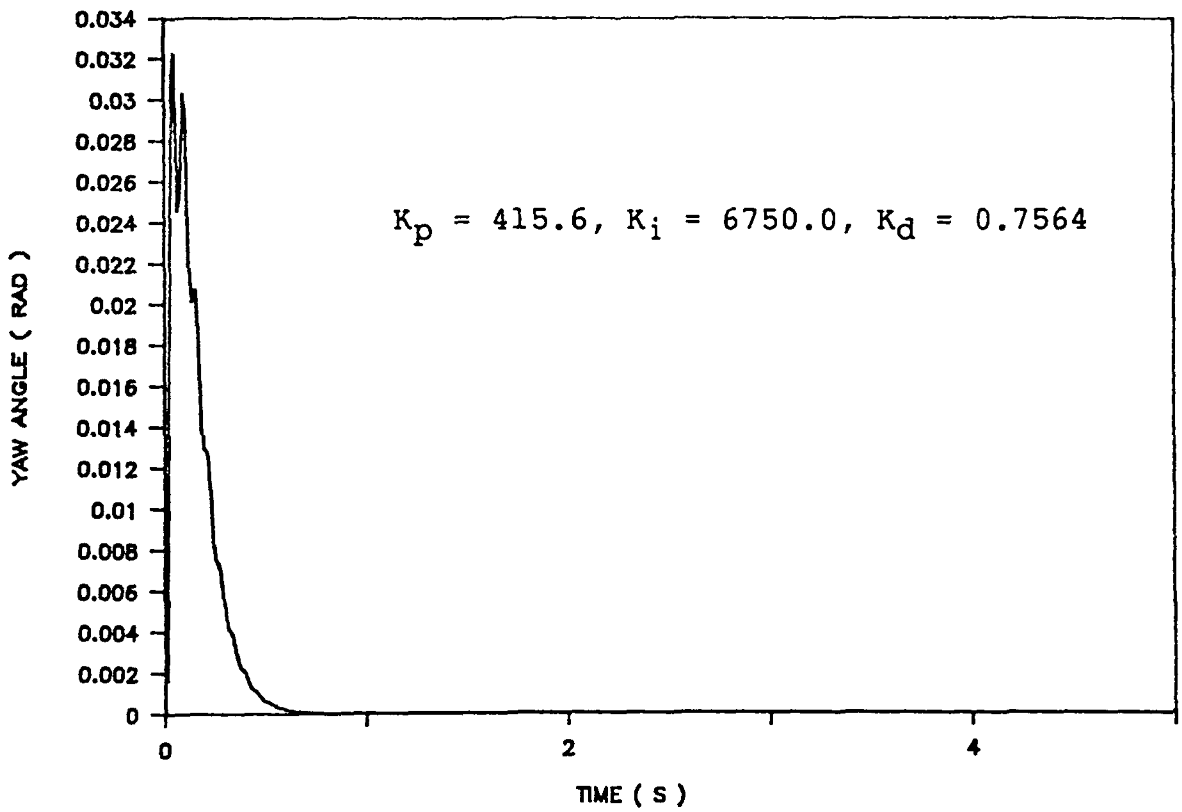


FIGURE 21. Step response with increasing control gains

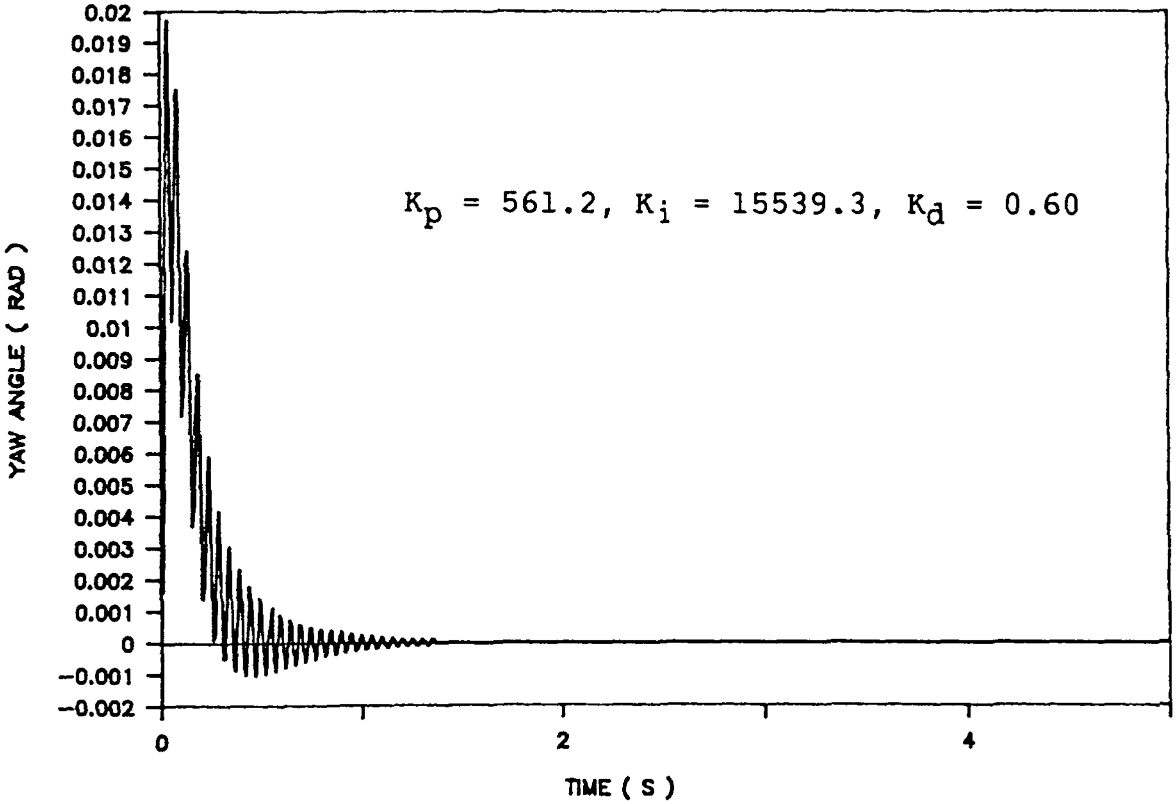


FIGURE 22. Step response with large K_i

During the continuous DO loop of the simulation procedure, the performance index ITSE decreases steadily from 9.278×10^{-5} to 2.43×10^{-7} before the run time failure caused by an asymptote action of the performance index function. Finally, a set of optimum control gains can be defined as following:

$K_p = 570.8$

$K_i = 32122.7$

$k_d = 1.62$

The related diagram is shown in Fig. 23. It can be seen that, although the performance index ITSE is minimized, the greater oscillation with the higher frequency appears in the step response. It also should be mentioned that the transient state begins to elongate along time axis when three gains have too large values.

It can be found that the optimization simulation based only on the criterion ITSE to choose three gains is unsatisfactory. In order to obtain a practically 'optimal' controller, other specifications such as percent overshoot, oscillation, and the settling time should also be introduced.

By investigating the performance index in the output of FCNMIN and the response curves in previous figures, it can be found that $K_p = 376$, $K_i = 2440$, and $K_d = 0.756$ satisfy the requirement proposed by those criteria except for a little oscillation. If K_p is increased from

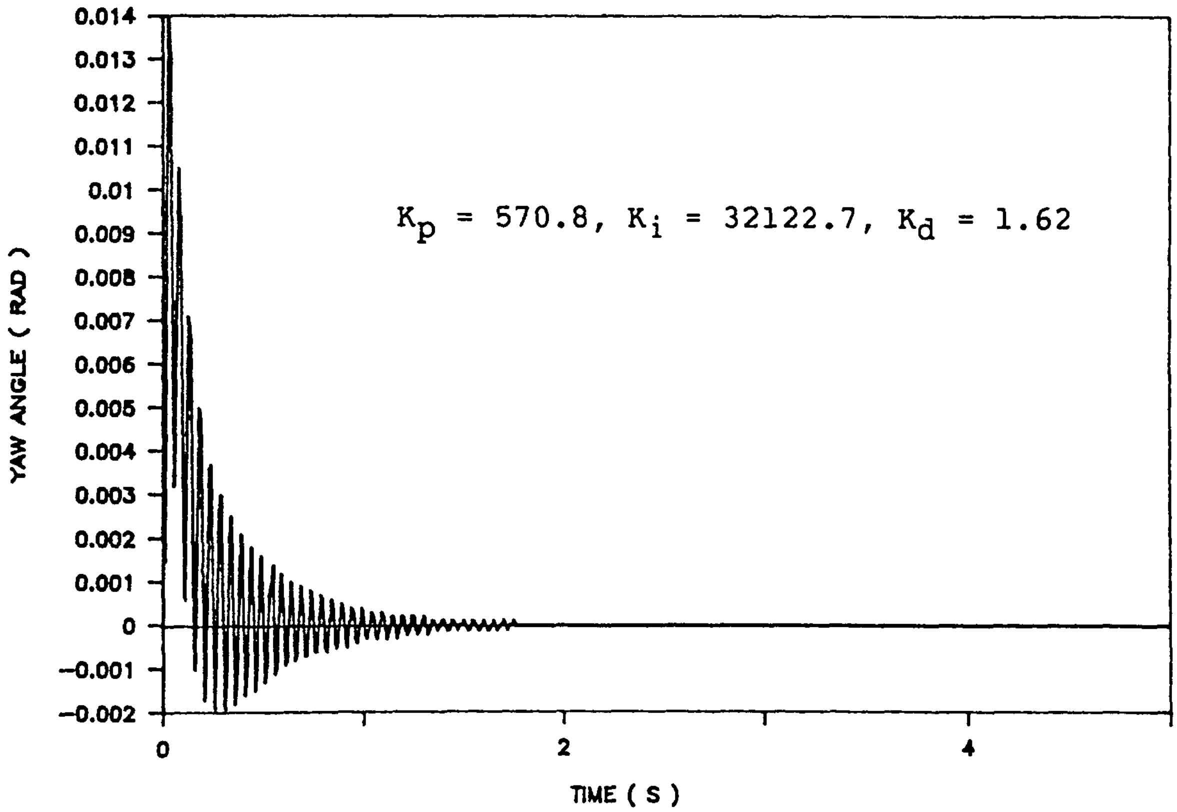


FIGURE 23. Step response with optimum control gains

376.0 to 400.0, K_i is reduced from 2440.0 to 2400.0, and K_d is raised from 0.756 to 1.0, the response oscillation will become negligibly small while the performance index is still satisfactory (Fig. 24).

Therefore, the practically 'optimal' control gains can be presented below:

$$K_p = 400.0$$

$$K_i = 2400.0$$

$$K_d = 1.0$$

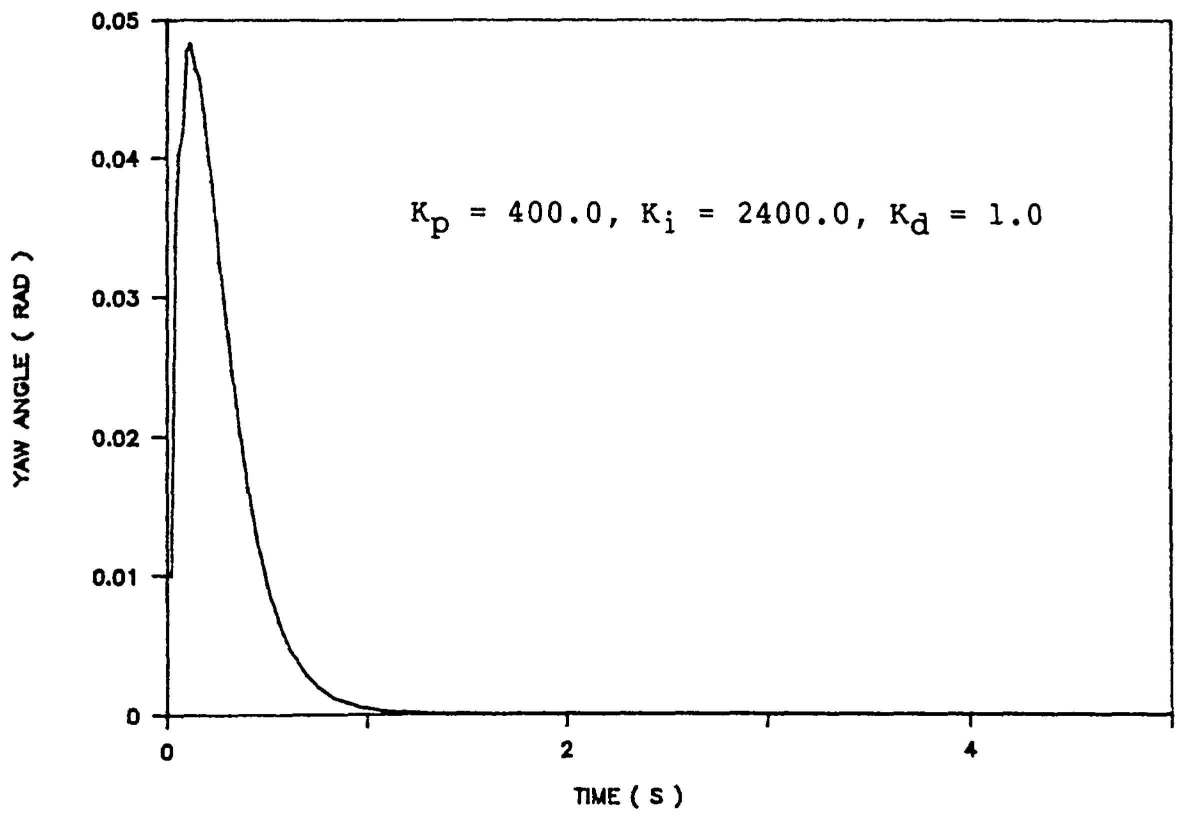


FIGURE 24. Step response with optimal control gains

TEST OF THE PID CONTROLLER ON A RANDOM INPUT

In practice, the random disturbing force is more realistic than the step input in the analysis of the directional behavior of the tractor system, especially for those tractors working in the field or on a side slope. Therefore, in this final simulation the random input is assumed to act on the C.G. of the tractor to test the 'goodness' of the optimal PID controller (K_p = 400, K_i = 2400, K_d = 1). A computer program called RANDOM.FOR has been used to generate a table of random forces between 0 and 1000 N (Appendix C). The graph of the random input forces is presented in Fig. 25.

A random input forcing function was treated as a series of pulses of uniform duration. The pulse that occurs at t_i after the system is disturbed from equilibrium may be regarded as two step functions, one of amplitude F_i starting at t_i and one of amplitude $-F_i$ starting at $t_i + \Delta t$. The Laplace transform of such a pulse may be obtained by applying the shifting theory

$$\mathscr{L}(\text{Pulse at } t_i) = e^{-t_i S} F_i (1-e^{-\Delta t S})/S$$

When a function in S-domain is multiplied by a delaying term e^{-DS}, the inverse transform is given by

$$\mathscr{L}^{-1}(e^{-DS}G(S)) = 0 \qquad \text{if } t < D$$

$$\mathscr{L}^{-1}(e^{-DS}G(S)) = g(t-D) \qquad \text{if } t \geq D$$

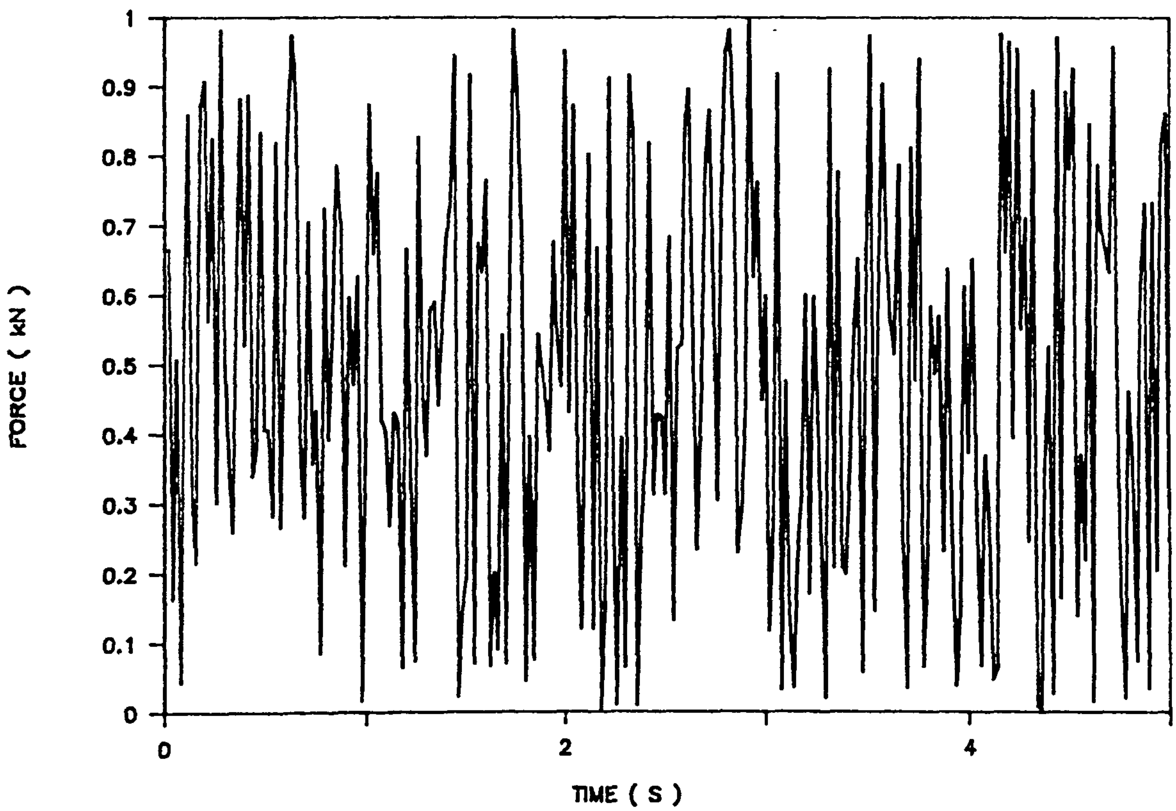


FIGURE 25. Simulated random input forces

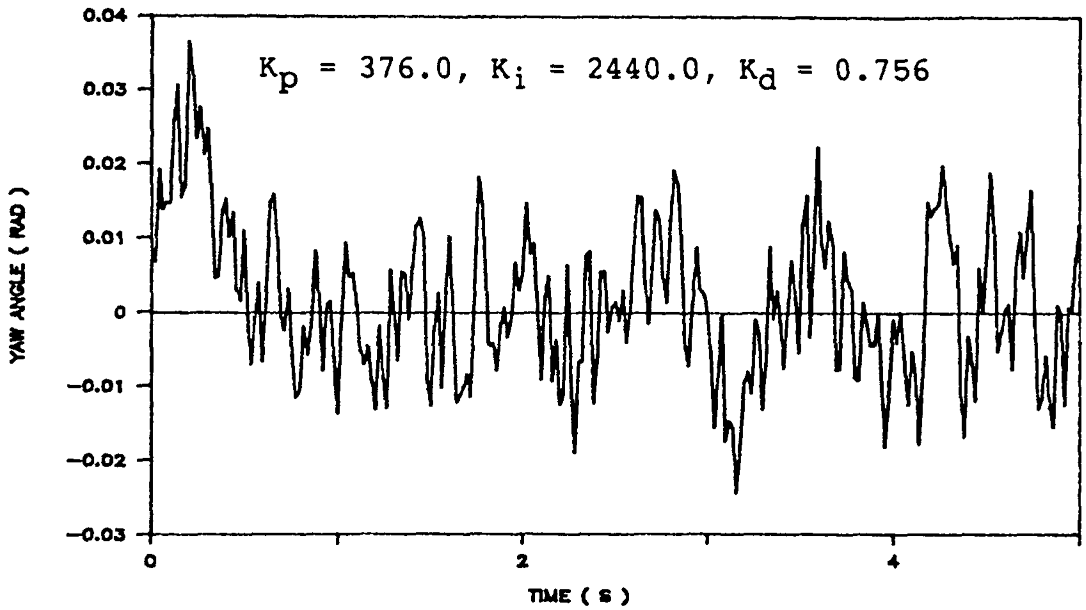
In this situation the function was: T(S)/S

where, T(S) = the closed-loop transfer function for the system.

The inversion of T(S)/S was handled in the computer algorithm by partial fractions. The conditions of 0 if $t < D$ and $g(t-D)$ if $t \geq D$ were handled using IF-THEN-ELSE statements. The FORTRAN source code for the algorithm is presented in Appendix A. Because the transfer function is linear, the response for the complete forcing function is just the sum of the effects of all the individual pulse responses. This approach was used because it is an exact solution that is true for all values of time. Consequently this solution is capable of showing oscillatory behavior during time periods less than that of an input pulse.

Fig. 26 has shown the effect of three control gains on the random response of the tractor. It is obvious that with the increase of three gains the response error and the oscillation amplitude is reduced but the oscillation frequency is increased.

Fig. 27 has indicated the yaw angle variation of the tractor under the control of the 'optimal' controller ($K_p = 400$, $K_i = 2400$, $K_d = 1$) travelling along a straight line within the time periods of 5 seconds. At first, an initial overshoot of $\psi(t)$ with the amplitude of 0.018 radian occurs at time t = 0.1 s. It rise to its maximum value 0.035 radian at t = 0.3 s. Then, $\psi(t)$ gradually approaches zero with some oscillation. After t = 0.5 s, the irregular oscillation around zero



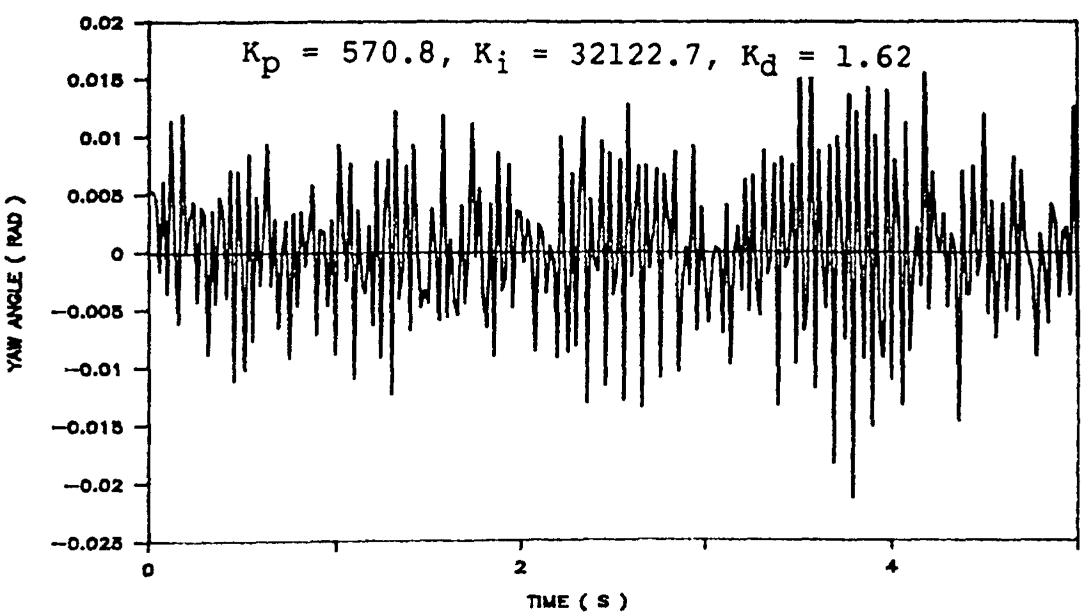


FIGURE 26. The effect of three gains on random response

continues and will never damp out as long as the random input exists, but the oscillation amplitudes have been kept within a reasonable range (-0.023 ~ +0.018 rad) due to the effective adjustment of the PID controller on the tractor yaw angle. The simulation has indicated that the 'optimal' controller selected above is able to control the tractor against the external disturbance and thus maintain the tractor on the preset path.

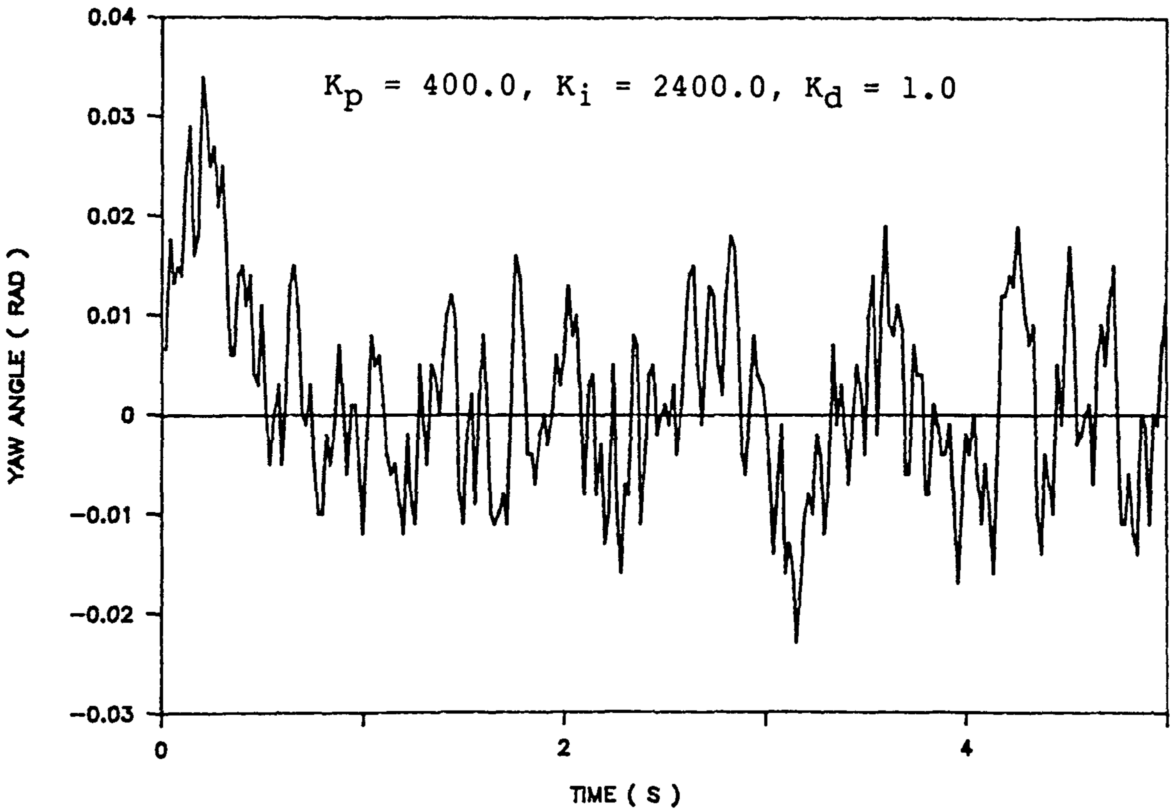


FIGURE 27. The test of optimal controller on random input

DISCUSSION AND CONCLUSIONS

A preliminary mathematical model describing the directional behavior of an agricultural four-wheel tractor has been developed and computer simulations for the time responses of the PID controller/tractor system to both step and random disturbing forces has been performed. This beginning work dealt only with the analysis and the simulation of the yaw angle of the bicycle tractor model moving on a level, hard surface. Figure 26 indicated that the maximum deviation of the yaw angle of the 'optimal' PID controller/tractor system, when subjected to random input, was about 2° and lasted about 0.1 second. It is also clear that before the yaw angle approached zero the average deviation of 1° was maintained for nearly 0.5 second. To some extent, the results in this thesis may be compared to those derived by James and Wierwille (1978), although their driver model is based on a nonlinear strategy model. Both operator/tractor systems have the same control principle of using yaw deviation error as input for control signal in lane keeping. The response graphs of two simulated system have a similar shape, except that the random response of the automatic controller/tractor system was more accurate. The simulation itself was not verified on a full scale tractor because one of the objectives of this investigation was to obtain information on responses so that later experimental work could be planned more rationally.

In order to overcome the limitation of the bicycle model in describing the lateral dynamics of the tractor, a multiple degree of freedom, full width model should be considered. The cornering stiffness of each tire of the tractor also plays an important role in the lateral dynamics of the tractor. There are several parameters that affect the cornering stiffness, such as weight transfer, inflation pressure, tire width and diameter, etc. A tractor is usually operating in field, so the tire/soil interaction should be introduced in the derivation of the mathematical model. Also further investigations of the lateral displacement Y should be undertaken. This study only considered the control of the yaw angle ψ and in a real situation control of Y would also be necessary.

From the various yaw angle response curves presented from Fig. 2 to Fig. 26, it can be seen that it is impossible for the PID controller to satisfy all of the performance specifications. For example, oscillation is difficult to eliminate only using the PID controller.

In general, from this beginning work it can be concluded that:

1. Basic understanding of the yaw angle behavior has been obtained by the analysis of the bicycle tractor model.
2. In the absence of any feedback, when the tractor is subjected to external disturbances, the tractor will leave the intended path and not return. It is technically stable because the roots of the characteristic equation remain in the left hand.
3. The PID controller/tractor system with K_p=400, K_i=2400, and K_d=1 is directionally stable and is fairly accurate in following a straight line.

REFERENCES

Bode, H. W. 1964. Feedback—The history of an idea. In Selected papers on mathematical trend in control theory. Dover, New York.

Brouhleit, G. 1925. La suspension de la direction de la voiture automobile, shimmy et dandinement. Societe des Ingenieurs Civils de France Bulletin 78:540-554.

Bundorf, R. T. and D. E. Pollock. 1963. Vehicle handling response to aerodynamic inputs. Trans. SAE 72:7-23.

Carson, J. M. and W. W. Wierwille. 1978. Development of a strategy model of the driver in lane keeping. Vehicle System Dynamics 7:233-253.

Chiesa, A. and L. Rinonapoli. 1967. Vehicle stability studied with a nonlinear seven degree model. Trans. SAE 76:1708-1724.

Dennis, J. E. and R. B. Schnabel. 1983. Numerical methods for unconstrained optimization and nonlinear equations. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Denny, C. D. and G. E. Rehkugler. 1974. Agricultural wheel-tractor overturns. Part I: Mathematical model. Trans. ASAE 17:477-492.

Ellis, J. 1963. The dynamics of vehicles during braking. Inst. of Mech. Engrs., Symposium on Control of Vehicles, London.

Evans, R. D. 1935. Properties of tires affecting riding, steering and handling. Trans. SAE 30:41-49.

Huston, J. C. and D. B. Johnson. 1982. Basic analytical results for lateral stability of car-trailer systems. Trans. SAE 91:560-569.

Jenkins, M. A. and J. E. Traub. 1970. A three-stage algorithm for real polynomials using quadratic iteration. SIAM J. NUMER. ANAL. 7, No. 4:545-566.

Kondo, M. and A. Ajimine. 1968. Driver sight point and dynamics of the driver vehicle system related to it. SAE Paper No. 680104.

Larson, D. L. 1971. Simulation of sideways overturning of wheel tractors on the side slopes. SAE Paper No. 710709.

MacAdam, C. C. and P. S. Fancher. 1986. A study of the closed-loop directional stability of various commercial vehicle configurations. Vehicle System Dynamics 15:367-382.

McRuer, D. T. and E. S. Krendel. 1974. Mathematical model of human pilot behavior. N. A. T. D. AGARDograph No. 188.

Metzger, M. 1985. Deadbeat control by PID digital controller. Control-Theory and Advanced Technology 1:275-286.

Moldenhauer, P. M. 1985. Vehicle lateral stability: Analytical relationship. Ph.D. dissertation. Iowa State University, Ames, Iowa.

Palm, W. J. 1983. Modelling, analysis, and control of dynamic system. John Wiley & Sons, Inc., New York.

Rehkugler, G. E. 1980. Simulation of articulated steer four-wheel drive agricultural tractor motion and overturns. Trans. ASAE 23:2-13.

Schafer, R. L. and R. E. Young. 1979. An automatic guidance system for tractors. Trans. ASAE 22:46-56.

Schilling, R. 1953. Directional control of automobiles. Industrial Mathematics 3:63-77.

Speckhart, F. H. 1973. A computer simulation for three dimensional vehicle dynamics. SAE Paper No. 730526.

Stonex, K. A. 1941. Car control factors and their measurements. Trans. SAE 36:81-93.

Thaler, G. J. 1974. Automatic control: Classical linear theory. Sect. II-E. Dowden, Hutchinson and Ross, Inc. Stroudsburg, Pa.

Wong, J. Y. 1978. Theory of ground vehicles. John Wiley & Sons, Inc., New York.

Xie, L. 1984. Simulation of tractor-trailer system stability. Master's thesis. Iowa State University, Ames, Iowa.

ACKNOWLEDGEMENTS

The author wishes to thank Dr. Richard J. Smith for invaluable advice, helpful suggestions and for serving as major professor.

Special thanks to Dr. Paul W. Claar II for helpful assistance and for serving as a committee member.

Sincere thanks to Dr. Jeffrey C. Huston for teaching me vehicle dynamics and for serving as a committee member.

APPENDIX A

```
C            PROGRAM  LSMAIN.FOR

C  THIS FORTRAN PROGRAM EVALUATES THE TIME RESPONSE OF A
C  CLASSICAL BICYCLE MODEL.  THE MODEL CONSIDERS THE EFFECT
C  OF EXTERNAL DISTURBANCES ON THE DIRECTIONAL STABILITY OF
C  THE DYNAMIC SYSTEM.

C  GLOSSARY OF VARIABLES
C  ---------------------

C    INTEGER

C      I           GENERAL DO VARIABLE
C      J           GENERAL COUNTING VARIABLE
C      M           DO VARIABLE
C      N           DO VARIABLE
C      INVRPT( )   INTEGER ARRAY COUNTING THE NUMBER OF
C                   REPEATED VALUES OF A MEMBER OF FACTORS
C      ISWIT       CONDITION VARIABLE
C      NDENM       ORDER OF DENOMINATOR POLYNOMIAL D(S)
C      NFACT       NUMBER OF FACTORS OF DENOMINATOR D(S)
C      NFACT1      NUMBER OF FACTORS OF MODIFIED D(S)
C      NIND        ORDER OF INPUT DENOMINATOR POLYNOMIAL
C      NNUMR       ORDER OF NUMERATOR POLYNOMIAL N(S)
C      NOUTD       ORDER OF OUTPUT DENOMINATOR POLYNOMIAL
C      NOUTN       ORDER OF OUTPUT NUMERATOR POLYNOMIAL
C      NREPT( )    INTEGER ARRAY RELATING TO THE NUMBER OF
C                    REPEATED VALUES IN THE ARRAY DFACTS( )
C      NSUB        THE NUMBER OF TIME SUBINTERVALS
C      NTEMP       ORDER OF TEMPORARY STORAGE POLYNOMIAL
C                  PTEMP( )

C    REAL

C      F( )        TIME RESPONSE OF THE LINEAR SYSTEM
C      G           RECIPROCAL OF ONE FACTORIAL
C      PNUMR( )    COEFFICIENTS OF NUMERATOR POLYNOMIAL N(S)
C      PIND( )     COEFFICIENTS OF INPUT DENOMINATOR POLYNOMIAL
C      POUTN( )    COEFFICIENTS OF OUTPUT NUMERATOR POLYNOMIAL
C      POUTD( )    COEFFICIENTS OF OUTPUT DENOMINATOR POLYNOMIAL
C      PTEMP( )    COEFFICIENTS OF TEMPORARY STORAGE POLYNOMIAL
C      T( )        TIME VARIABLE ARRAY
C      TIME        TOTAL TIME LENGTH
C      U( )        RANDOM NUMBERS IN THE RANGE 0 AND 1
C      Y( )        ONE-D VARIABLE ARRAY

C    DOUBLE PRECISION
```

```
C      PDENM( )   COEFFICIENTS OF DENOMINATOR POLYNOMIAL D(S)
C      S          ESTIMATION OF PERFORMANCE INDEX  ITSE
C      ZEROR, ZEROI   OUTPUT VECTORS OF REAL AND IMAGINARY
C                     PARTS OF THE ZEROS OF D(S)

C    COMPLEX

C      CFF( )     ARRAY OF PARTIAL FRACTION COEFFICIENTS
C      CTEMP( )   WORKING ARRAY OF THE RESULTING POLYNOMIAL
C                   COEFFICIENTS IN COMPLEX FORM
C      DFACTS( )  ARRAY OF DENOMINATOR POLYNOMIAL FACTORS
C      DFACT1( )  ARRAY OF MODIFIED DENOMINATOR POLY. FACTORS
C      DROOT      DROOT = CMPLX ( ZEROR(I), ZEROI(I) )

C    LOGICAL

C      FAIL       OUTPUT LOGICAL PARAMETER, TRUE IF ONLY
C                 LEADING COEFFICIENT IS ZERO OR IF SUBROUTINE
C                 RPOLY HAS FOUND FEWER THAN NDENM ZEROS.  IN
C                 THE LATTER CASE NDENM IS RESET TO THE NUMBER
C                 OF ZEROS FOUND.

C  SUBPROGRAMS CALLED
C  ------------------

C    RPOLY ( PDENM, NDENM, ZEROR, ZEROI, FAIL )
C      THIS SUBROUTINE ACCEPTS THE REAL COEFFICIENTS
C      PDENM( ) OF DENOMINATOR POLYNOMIAL D(S) AND RETURNS
C      THE ZEROS OF D(S).

C    CSORT ( NDENM, DFACTS, TOL, TOLZ )
C      THIS SUBROUTINE ACCEPTS A COMPLEX ARRAY
C      DFACTS( ) AND RETURNS THE CORRESPONDING SORTED AND ADJUSTED
C      COMPLEX ARRAY IN PLACE OF THE ORIGINAL ARRAY DFACTS( ).

C    FCTCNT ( NDENM, DFACTS, NREPT, INVRPT )
C      THIS SUBROUTINE COUNTS THE NUMBER OF REPEATED FACTORS
C      AND STORES THE NUMBER IN THE INTEGER ARRAYS NREPT( )
C      AND INVRPT( ).

C    PRTFRC ( NNUMR, PNUMR, NDENM, DFACTS, J, CFF )
C      THIS SUBROUTINE EVALUATES THE PARTIAL FRACTION
C      COEFFICIENTS OF Y(S) = N(S) / D(S) WHEN THE DFACTS( )
C      ARE NON-REPEATED.

C    FCTPLY ( NFACT1, DFACT1, NIND, PIND, CTEMP )
C      THIS SUBROUTINE ACCEPTS A COMPLEX ARRAY DFACT1( ) AND
C      RETURNS A REAL ARRAY OF POLYNOMIAL COEFFICIENTS PIND( ).
```

```
C    DFQUOT ( NNUMR, PNUMR, NIND, PIND, NOUTN, POUTN, NOUTD,
C                                     POUTD, NTEMP, PTEMP )
C      THIS SUBROUTINE ACCEPTS TWO REAL POLYNOMIAL
C      COEFFICIENT ARRAYS, PNUMR( ) AND PIND( ), OF THE
C      NUMERATOR AND DENOMINATOR OF A QUOTIENT, AND RETURNS
C      THE NUMERATOR AND DENOMINATOR COEFFICIENTS ARRAYS OF
C      THE DIFFERENTIAL QUOTIENT, POUTN( ) AND POUTD( ).

C    COMPLEX FUNCTION CPEVAL ( NIN, PIN, X )
C      THIS IS A COMPLEX FUNCTION THAT ACCEPTS A VALUE OF A
C      COMPLEX ARGUMENT X AND RETURNS THE VALUE OF A REAL
C      POLYNOMIAL.

C    REVERSE( NDENM, PDENM )
C      THIS SUBROUTINE ACCEPTS THE COEFFICIENTS PDENM( )
C      OF D(S) IN ASCENDING ORDER AND RETURNS THE REVERSED
C      PDENM( ) IN PLACE OF ORIGINAL PDENM( ).

C    RESPONSE( CFF, DFACTS, INVRPT, NSUB, NDENM, TIME, F, T )
C      THIS SUBROUTINE ACCEPTS THE COEFFICIENTS CFF( ),
C      DFACTS( ) AND TIME. THEN IT RETURN THE VALUE OF
C      TIME RESPONSE ARRAY F( ) AND CORRESPONDING TIME T( ).

C    RESPNS( CFF, DFACTS, INVRPT, NSUB, NDENM, TIME, Y )
C      THIS SUBROUTINE  HAS THE SIMILAR FUNCTION AS
C      RESPONSE(...).

C    SIMPSON( NSUB, TIME, T, Y, S )
C      THIS SUBROUTINE IS USED TO ESTIMATE THE AREA UNDER
C      A CURVE BY USING SIMPSON'S RULE.

C    FUNCTION RAND( L )
C      THIS FUNCTION GENERATES A SET OF RANDOM NUMBERS
C      BETWEEN 0 AND 1, INCLUDING THE END POINTS, AND
C      RETURNS THE RANDOM NUMBERS IN THE FILE NAMED DATA.
C    _______________________________________________________

C                         MAIN   PROGRAM

C    _______________________________________________________

      COMMON /SMROOT/IDS, NTRY
      INTEGER  IDS, NTRY

      INTEGER  I,J,M,N,INVRPT(10),ISWIT,NDENM,NFACT1,NSUB
      INTEGER  NIND, NNUMR, NOUTN, NOUTD, NREPT(10), NTEMP

      REAL     G, PNUMR(11), PIND(10), POUTD(10), POUTN(10)
      REAL     PTEMP(10), T(1000), TIME, TOL, TOLZ
      REAL     F(1000), U(1000), Y(1000)
```

```
      DOUBLE PRECISION  PDENM(11), S, ZEROR(10), ZEROI(10)

      COMPLEX  CFF(10),CTEMP(11),DFACTS(10),DFACT1(10),DROOT
      COMPLEX  CPEVAL

      LOGICAL  FAIL

      INTEGER  NPTS, ISYM, MODE
      REAL     TSIZE, TSF, TMIN, YSIZE, YSF, YMIN
      CHARACTER*20  TLAB, YLAB, GLAB, DATLAB

      IDS=14
      NTRY=0

      OPEN(UNIT=IDS,FILE='SIMPLT.DAT',STATUS='UNKNOWN.')
      CLOSE(UNIT=IDS,STATUS='DELETE')
      OPEN(UNIT=1, FILE='DATA', STATUS='OLD')
      OPEN(UNIT=2, FILE='OUTPUT.DAT', STATUS='NEW')



      WRITE(*,*) 'IF STEP INPUT, ENTER ISWIT=1,'
      WRITE(*,*) 'IF RANDOM INPUT, ENTER ISWIT=2'
      READ(*,*) ISWIT
      WRITE(*,*) 'ENTER THE ORDER OF NUMERATOR N(S)'
      READ(*,*) NNUMR
      WRITE(2,666) NNUMR
666   FORMAT(1X, 'NNUMR = ', I2)
      WRITE(2,*)
      WRITE(*,*) 'ENTER COEFF. OF NUMERATOR, CONSTANT FIRST'
      READ(*,*) ( PNUMR(I), I=1, NNUMR+1 )
      WRITE(2,667) ( I, PNUMR(I), I=1,NNUMR+1 )
667   FORMAT(1X, 'PNUMR(',I2,')=', F20.7)
      WRITE(2,*)
      WRITE(*,*) 'ENTER ORDER OF DENOMINATOR D(S)'
      READ(*,*) NDENM
      WRITE(2,668) NDENM
668   FORMAT(1X, 'NDENM = ', I2)
      WRITE(2,*)
      WRITE(*,*) 'ENTER COEFF. OF DENOMINATOR,CONSTANT FIRST'
      READ(*,*) ( PDENM(I), I=1, NDENM+1 )
      WRITE(2,669) ( I, PDENM(I), I=1, NDENM+1 )
669   FORMAT(1X,'PDENM(',I2,')=', F15.7)
      WRITE(2,*)

      WRITE(*,*) 'ENTER TOTAL TIME LENGTH'
      READ(*,*) TIME
      WRITE(2,700) TIME
700   FORMAT(1X, 'TIME=', F8.4)
```

```
      WRITE(2,*)

      WRITE(*,*) 'ENTER THE NUMBER OF TIME SUBINTERVALS'
      READ(*,*) NSUB
      WRITE(2,701) NSUB
701   FORMAT(1X, 'NSUB=', I3)
      WRITE(2,*)

C  REVERSE THE COEFFICIENT ARRAY PDENM( ) OF D(S)

      CALL REVERSE(NDENM,PDENM)

C  CALCULATE THE ZEROS OF REAL DENOMINATOR POLYNOMIAL D(S)

      CALL RPOLY(PDENM, NDENM, ZEROR, ZEROI, FAIL)
      DO 10 I=1,NDENM
         WRITE(2,702) I, ZEROR(I), I, ZEROI(I)
702      FORMAT(1X,'ZEROR(',I2,')=',F15.7,2X,'ZEROI(',I2,
     $          ')=', F15.7)
10    CONTINUE

C  FACTORIZE THE DENOMINATOR POLYNOMIAL D(S)

      DO 20 I=1, NDENM
        DFACTS(I)=CMPLX(-ZEROR(I),-ZEROI(I))
20    CONTINUE


C  SORT AND ADJUST DFACTS(I) BY ASCENDING ORDER

      TOL=1.0E-04
      TOLZ=1.0E-06
      CALL CSORT(NDENM, DFACTS, TOL, TOLZ)
      WRITE(2,*)
      DO 30 I=1,NDENM
         WRITE(2,703) I, DFACTS(I)
703      FORMAT(1X, 'DFACTS(',I2,')=', 2F15.7)
30    CONTINUE

      WRITE(2,*)

C  FIND THE REPEATED FACTORS IN THE DENOMINATOR D(S)

      CALL FCTCNT(NDENM, DFACTS, NREPT, INVRPT)
      DO 40 I=1,NDENM
         WRITE(2,705) I, INVRPT(I)
705      FORMAT(1X, 'INVRPT(',I2,')=', I2)
40    CONTINUE

      WRITE(2,*)
```

```
C  EVALUATE THE COEFF. OF PARTIAL FRACTION Y(S)=N(S)/D(S)

       J=0
1000   CONTINUE

       J=J+1
       IF(INVRPT(J).EQ.1) THEN

C      COEFF. OF PARTIAL FRACTION FOR NON-REPEATED FACTORS

           CALL PRTFRC(NNUMR, PNUMR, NDENM, DFACTS, J, CFF)

           WRITE(2,707) J, CFF(J)
707        FORMAT(1X, 'CFF(',I2,')=', 2F15.7)

       ELSE

C        FIND THE FACTOR ARRAY OF MODIFIED D(S)

           DO 50 I=1,J-1
50            DFACT1(I)=DFACTS(I)
           DO 60 I=J, NDENM-INVRPT(J)
60            DFACT1(I)=DFACTS(I+INVRPT(J))

C        CALCULATE THE COEFFICIENTS OF MODIFIED D(S)

           NFACT1=NDENM-INVRPT(J)

           CALL FCTPLY(NFACT1,DFACT1,NIND,PIND,CTEMP)

C      COEFF. OF PARTIAL FRACTION FOR REPEATED FACTORS

           DROOT=-DFACTS(J)
           CFF(J)=CPEVAL(NNUMR,PNUMR,DROOT)/
     $            CPEVAL(NIND,PIND,DROOT)
           G=1.0
           DO 90 I=J+1, INVRPT(J)+(J-1)
              G=G/(I-J)

              CALL DFQUOT(NNUMR,PNUMR,NIND,PIND,NOUTN,
     $                    POUTN,NOUTD,POUTD,NTEMP,PTEMP)

              DROOT=-DFACTS(J)
              CFF(I)=G*CPEVAL(NOUTN,POUTN,DROOT)/
     $                 CPEVAL(NOUTD,POUTD,DROOT)
              NNUMR=NOUTN
              NIND=NOUTD

              DO 70 M=1,NOUTN
```

```
70             PNUMR(M)=POUTN(M)
            DO 80 N=1,NOUTD
80             PIND(N)=POUTD(N)
90        CONTINUE

            DO 100 K=J, INVRPT(J)+(J-1)
               WRITE(2,709) K, CFF(K)
709            FORMAT(1X, 'CFF(',I2,')=', 2F15.7)
100         CONTINUE

          J=INVRPT(J)+(J-1)

         ENDIF

         IF (J.LT.NDENM) GOTO 1000

C  CALCULATE THE TIME RESPONSE F( ) OF THE BICYCLE MODEL

C    CALCULATE THE STEP RESPONSE F( ) OF THE LINEAR SYSTEM

      IF(ISWIT.EQ.1) THEN

         CALL RESPONSE(CFF,DFACTS,INVRPT,NSUB,NDENM,
    $                 TIME,F,T)
         DO 120 I=1,NSUB
            WRITE(2,801) I, F(I), I, T(I)
801         FORMAT(1X,'F(',I3,')=',F15.7,5X,'T(',I3,
    $              ')=', F8.4)
120      CONTINUE

C    EVALUATE PERFORMANCE INDEX ITSE USING SIMPSON'S RULE

         CALL SIMPSON(NSUB, TIME, T, F, S)

         WRITE(2,*)
         WRITE(2,803) S
803      FORMAT(1X, 'PERFORMANCE INDEX ITSE=', D20.7)

      ELSE

C    COMPUTE RANDOM RESPONSE F( ) OF THE LINEAR SYSTEM

         CALL RESPNS(CFF,DFACTS,INVRPT,NSUB,NDENM,TIME,Y)

         DO 130 I=1,100
            N=(I-1)*8
            READ(1,805) (U(M), M=N+1,N+8)
805         FORMAT(1X,8F8.4)
130      CONTINUE
```

```
            DO 150 K=1,NSUB
               SUM=0.0
               DO 140 I=0,K-1
140               SUM=SUM+U(K-I)*Y(I+1)
               F(K)=SUM
               T(K)=TIME*K/NSUB
               WRITE(2,807) K, F(K), K, T(K)
807            FORMAT(1X,'F(',I3,')=',F15.7,5X,'T(',I3,
     $                ')=', F8.4)
150         CONTINUE

         ENDIF

C        SIMPLOTTER FOR THE GRAPH OF THE TIME RESPONSE

            NPTS=NSUB
            ISYM=1
            MODE=3
            TSIZE=5.0
            TSF=TIME/5.0
            TMIN=0.0
            YSIZE=4.0
            YSF=0.0
            YMIN=0.0
            TLAB='TIME'
            YLAB='RESPONSE'
            GLAB='LINEAR SYSTEM RESPONSE'
            DATLAB='    '

            CALL GRAPH (NPTS,T,F,ISYM,MODE,TSIZE,YSIZE,TSF,
     $                  TMIN,YSF,YMIN,TLAB,YLAB,GLAB,DATLAB)

         CLOSE(UNIT=2)
         CLOSE(UNIT=1)
         CLOSE(IDS)

         STOP

         END


         COMPLEX FUNCTION CPEVAL (NIN, PIN, X)

         INTEGER  I, NIN
         REAL     PIN(1)
         COMPLEX  X

         CPEVAL=CMPLX(0.0)

         DO 5 I=(NIN+1),2,-1
```

```
      CPEVAL=X*(CMPLX(PIN(I))+CPEVAL)
5     CONTINUE
      CPEVAL=CPEVAL+CMPLX(PIN(1))
      RETURN
      END
C...V....V....V....V....V....V....V....V....V....V....V....V
      SUBROUTINE CSORT (NSIZE,DFACTS,TOL,TOLZ)
C
C     GLOSSARY OF VARIABLES
C     ---------------------
C     INTEGER
C     I          GENERAL COUNTING VARIABLE
C     ILOW       INDEX LOCATING THE LOWER BOUND OF AN IDENTICAL
C                GROUP OF PRIMARY ITEMS THAT ARE BEING SORTED
C                ACCORDING TO A SECONDARY CHARACTERISTIC.
C     IHIGH      INDEX LOCATING THE UPPER BOUND OF AN IDENTICAL
C                GROUP OF PRIMARY ITEMS THAT ARE BEING SORTED
C                ACCORDING TO A SECONDARY CHARACTERISTIC.
C     J          SECONDARY LEVEL GENERAL COUNTING VARIABLE
C     NSIZE      NUMBER OF MEMBERS IN THE COMPLEX ARRAY DFACTS
C
C     REAL
C     AITMP1     IMAGINARY PART OF DFACTS(I)
C     AITMP2     IMAGINARY PART OF DFACTS(I+1)
C     RTEMP1     REAL PART OF DFACTS(I)
C     RTEMP2     REAL PART OF DFACTS(I+1)
C     TOL        TOLERANCE USED TO DETERMINE IF THE PARTS OF TWO
C                FACTORS ARE SET EQUAL
C     TOLZ       TOLERANCE USED TO DETERMINE IF THE PARTS OF TWO
C                FACTORS ARE SET EQUAL TO ZERO
C
C     COMPLEX
C     CTEMP1     TEMPORARY VARIABLE USED DURING BUBBLE SORTING
C     DFACTS()   COMPLEX ARRAY BEING SORTED, THE SORTED ARRAY IS
C                RETURNED IN PLACE OF DFACTS.
C
C     LOGICAL
C     LSWAP      FLAG USED TO INDICATE IF A SWAP HAS TAKEN PLACE
C                DURING A SORT, LSWAP = .FALSE. WHEN NO SWAP HAS
C                TAKEN PLACE.
C
C     ALGORITHM DESCRIPTION
C     ---------------------
```

```
C     THIS SUBROUTINE TAKES A COMPLEX ARRAY AND SORTS THIS IN
C     A PRIMARY CATEGORY OF ASCENDING VALUES OF THE IMAGINARY
C     PARTS.  THE SECONDARY SORT IS BY ASCENDING ORDER OF REAL
C     PARTS.  A DOUBLE SORTING METHOD IS USED.
C     IN GENERAL IN AUTOMATIC CONTROL SITUATIONS, ACCURACIES
C     BETTER THAN ABOUT 1 PART IN 1.0E+05 ARE NOT REQUIRED.
C     SECTIONS IN THIS CODE EXAMINE THE MAGNITUDE OF BOTH
C     THE REAL AND IMAGINARY PARTS OF EACH FACTOR.  IF THESE
C     MAGNITUDES ARE LESS THAN OR EQUAL TO A TOLERANCE TOLZ,
C     THEN THE PART IS SET EXACTLY TO ZERO.  IN A SIMILAR
C     FASHION, IF TWO REAL OR TWO IMAGINARY PARTS HAVE THE
C     SAME MAGNITUDE WITHIN A RELATIVE ERROR TOLERANCE OF TOL,
C     THEN THE PARTS ARE SET EXACTLY EQUAL.  THESE PROCEDURES
C     ARE UNDERTAKEN TO ALLOW MULTIPLE FACTORS TO BE CLEARLY
C     DETECTED.  THESE PROCEDURES ARE CONSIDERED JUSTIFIED
C     FROM EXPERIENCE WITH POLYNOMIAL ROOT FINDING ROUTINES.
C     THESE ARE SELDOM CAPABLE OF RETURNING "EXACT" ROOTS
C     EVEN IF SUPPLIED WITH DATA KNOWN TO YIELD MULTIPLE ROOTS.
C     IF-THEN-ELSE LOOPS ARE USED THROUGHOUT TO AVOID JUMPING
C     OUT OF DO LOOPS.

      INTEGER I,ILOW,IHIGH,J,NSIZE
      REAL AITMP1,AITMP2,RTEMP1,RTEMP2,TOL,TOLZ
      COMPLEX DFACTS(NSIZE),CTEMP1
      LOGICAL LSWAP
C
C     THIS SECTION OF THE ALGORITHM EXAMINES THE REAL AND
C     IMAGINARY PARTS OF EACH FACTOR.  IF THE PART IS LESS
C     THAN A PRESET TOLERANCE TOLZ THEN THE PART IS SET TO BE
C     EXACTLY ZERO.

      DO 5 I=1,NSIZE
C
      AITMP1=AIMAG(DFACTS(I))
      RTEMP1=REAL(DFACTS(I))
C
C     TEST IF THE REAL PART OF DFACTS(I) IS SUFFICIENTLY CLOSE
C     TO ZERO THAT IT CAN BE SET EXACTLY EQUAL TO ZERO
C
        IF (ABS(RTEMP1) .LE. TOLZ) THEN
C
        RTEMP1=0.0
C
        END IF
C
C       TEST IF THE IMAGINARY PART OF DFACTS(I) IS SUFFI-
C       CIENTLY CLOSE TO ZERO THAT IT CAN BE SET EXACTLY EQUAL
C       TO ZERO.

        IF (ABS(AITMP1) .LE. TOLZ) THEN
```

```
C
        AITMP1=0.0
C
        END IF
C
        DFACTS(I)=CMPLX(RTEMP1,AITMP1)
C
    5 CONTINUE
C
C     THE NEXT PART OF THE SUBROUTINE SORTS THE ARRAY DFACTS
C     IN ASCENDING ORDER OF THE IMAGINARY PARTS.
C
   10 CONTINUE
C
      I=1
      LSWAP=.FALSE.
C
   20 CONTINUE
C
      CTEMP1=DFACTS(I)
      AITMP1=AIMAG(DFACTS(I))
      RTEMP1=REAL(DFACTS(I))
      AITMP2=AIMAG(DFACTS(I+1))
      RTEMP2=REAL(DFACTS(I+1))
C
C     TEST THAT AITMP2 IS ZERO TO AVOID A DIVIDE-BY-ZERO IN
C     THE APPROXIMATE EQUALITY CHECK THAT FOLLOWS.
C
      IF (AITMP2 .EQ. 0.0) THEN
C
C     TEST IF AITMP1 AND AITMP2 ARE SUFFICIENTLY CLOSE IN
C     MAGNITUDE THAT THEY SATISFY THE RELATIVE ERROR CONDITION.
C     NOTE THAT THIS TEST WILL DISCRIMINATE BETWEEN COMPLEX
C     CONJUGATES BECAUSE THE ABSOLUTE VALUE OF AITMP1/AITMP2
C     IS NOT TAKEN.

      ELSE IF(ABS(1.0-AITMP1/AITMP2) .LE. TOL) THEN
C
C     IF TRUE, MATCH THE COMPLEX PARTS OF DFACTS(I) AND
C     DFACTS(I+1)

      DFACTS(I+1)=CMPLX(RTEMP2,AITMP1)
C
      END IF
C
C     TEST IF AITMP1 > AITMP2 AND SWITCH THE ORDER OF
C     DFACTS(I) AND DFACTS(I+1)
C
      IF (AITMP1 .GT. AITMP2) THEN
C
```

```
      DFACTS(I)=DFACTS(I+1)
      DFACTS(I+1)=CTEMP1
      LSWAP=.TRUE.
C
      END IF
C
      I=I+1
C
      IF (I .LT. NSIZE) THEN
C
C     IF THE COMPLETE RANGE HAS NOT YET BEEN EXAMINED,
C     RETURN TO THE POINT AND REPEAT THE SIZE TEST.
C
      GO TO 20
C
      ELSE IF (LSWAP) THEN
C
C     THE ALGORITHM REACHES THIS POINT WHEN THE COMPLETE RANGE
C     HAS BEEN TESTED.  IF A SWAP HAS OCCURRED (I.E. LSWAP =
C     .TRUE.), THEN THE ALGORITHM RETURNS TO THE POINT WHERE
C     THE COUNTING VARIABLE IS INITIALIZED AND A FURTHER
C     BUBBLE THROUGH THE COMPLETE RANGE IS PERFORMED.
C
      GO TO 10
C
      END IF
C
C     THE NEXT PART OF THE SUBROUTINE SORTS THE ARRAY DFACTS
C     IN ASCENDING ORDER OF THE REAL PARTS WITHIN EACH SET OF
C     IDENTICAL IMAGINARY PARTS.

      I=1
C
   30 CONTINUE
C
C     FIND THE LOWER (ILOW) AND UPPER (IHIGH) INDICES OF
C     MEMBERS WITH THE SAME IMAGINARY PARTS.
C
      IF (AIMAG(DFACTS(I)) .EQ. AIMAG(DFACTS(I+1))) THEN
C
      ILOW=I
C
   40 CONTINUE
C
      I=I+1
      IHIGH=I
C
        IF ((AIMAG(DFACTS(I)) .EQ. AIMAG(DFACTS(I+1))) .AND.
     &  (I .LT. NSIZE)) THEN
C
```

```
C       CONTINUE INCREMENTING AND UPDATING IHIGH UNTIL THE
C       IMAGINARY PARTS DO NOT MATCH OR UNTIL THE END OF THE
C       LIST IS REACHED.

        GO TO 40
C
        ELSE
C
   50   CONTINUE
C
        J=ILOW
        LSWAP=.FALSE.
C
C       PERFORM A BUBBLE SORT BASED ON THE REAL PARTS OF THE
C       ARRAY BETWEEN THE INDICES ILOW AND IHIGH.
C
   60   CONTINUE
C
C
        CTEMP1=DFACTS(J)
        AITMP1=AIMAG(DFACTS(J))
        RTEMP1=REAL(DFACTS(J))
        AITMP2=AIMAG(DFACTS(J+1))
        RTEMP2=REAL(DFACTS(J+1))
C
C         TEST THAT RTEMP2 IS ZERO TO AVOID A DIVIDE-BY-ZERO
C         IN THE APPROXIMATE EQUALITY CHECK THAT FOLLOWS.
C
          IF (RTEMP2 .EQ. 0.0) THEN
C
C         TEST IF RTEMP1 AND RTEMP2 ARE SUFFICIENTLY CLOSE IN
C         MAGNITUDE THAT THEY SATISFY THE RELATIVE ERROR
C         CONDITION.  NOTE THAT THIS TEST WILL DISCRIMINATE
C         BETWEEN EQUAL REAL PARTS OF THE OPPOSITE SIGN BECAUSE
C         THE ABSOLUTE VALUE OF RTEMP1/RTEMP2 IS NOT TAKEN.
C
          ELSE IF(ABS(1.0-RTEMP1/RTEMP2) .LE. TOL) THEN
C
C         IF TRUE, MATCH THE REAL PARTS OF DFACTS(J) AND
C         DFACTS(J+1)

          DFACTS(J+1)=CMPLX(RTEMP1,AITMP2)
C
          END IF
C
C         TEST IF RTEMP1 > RTEMP2 AND SWITCH THE ORDER OF
C         DFACTS(J) AND DFACTS(J+1)
C
          IF (RTEMP1 .GT. RTEMP2) THEN
C
```

```
          DFACTS(J)=DFACTS(J+1)
          DFACTS(J+1)=CTEMP1
          LSWAP=.TRUE.
C
          END IF
C
          J=J+1
C
          IF (J .LT. IHIGH) THEN
C
C         IF THE RANGE HAS NOT YET BEEN EXAMINED, RETURN AND
C         REPEAT THE SIZE TEST.  THE COUNTING VARIABLE HAS
C         ALREADY BEEN INCREMENTED.
C
          GO TO 60
C
          ELSE IF (LSWAP) THEN
C
C         THE ALGORITHM REACHES THIS POINT WHEN THE COMPLETE
C         RANGE HAS BEEN TESTED.  IF A SWAP HAS OCCURRED (I.E.
C         LSWAP = .TRUE.), THEN THE ALGORITHM RETURNS TO THE
C         POINT WHERE THE COUNTING VARIABLE IS INITIALIZED AND
C         A FURTHER BUBBLE THROUGH THE RANGE IS PERFORMED.

          GO TO 50
C
          END IF
C
        END IF
C
      ELSE
C
      I=I+1
C
      END IF
C
      IF (I .LT. NSIZE) THEN
C
C     IF THE COMPLETE RANGE HAS NOT YET BEEN EXAMINED,
C     RETURN AND REPEAT THE SORTING PROCEDURE.  THE COUNTING
C     VARIABLE HAS ALREADY BEEN INCREMENTED.
C
      GO TO 30
C
      END IF
C
      RETURN
      END

C...V....V....V....V....V....V....V....V....V....V....V....V
```

```
      SUBROUTINE FCTCNT (NFACT,DFACTS,NREPT,INVRPT)
C
C   GLOSSARY OF VARIABLES
C   ---------------------
C
C   INTEGER
C   I          LOOP COUNTING VARIABLE
C   IHIGH      UPPER INDEX OF A RANGE OF EQUAL VALUES IN DFACTS
C   ILOW       LOWER INDEX OF A RANGE OF EQUAL VALUES IN DFACTS
C   INVRPT()   ARRAY CONTAINING THE NUMBER OF REPEATED VALUES
C              OF A MEMBER OF DFACTS
C   NREPT()    ARRAY RELATING TO THE NUMBER OF REPEATED VALUES
C              IN THE ARRAY DFACTS
C
C   COMPLEX
C   DFACTS()   ARRAY OF COMPLEX VALUES THAT HAS BEEN SORTED
C              IN ASCENDING ORDER, FIRST BY IMAGINARY PART AND
C              THEN BY REAL PART WITHIN THE IMAGINARY PARTS
C
C   LOGICAL
C   MATCH      TRUE WHEN TWO ADJACENT VALUES OF DFACTS ARE
C              RECOGNIZED AS THE SAME
C
C   ALGORITHM DESCRIPTION
C   ---------------------
C     THIS SUBROUTINE RECOGNIZES REPEATED VALUES IN AN ARRAY
C     OF COMPLEX NUMBERS, THE FACTORS OF A POLYNOMIAL, AND
C     ANNOTATES A COMPANION ARRAY, NREPT, WITH:-
C        0      VALUE IS NOT REPEATED
C        1      FIRST OCCURRENCE OF REPEATED VALUE
C        2      SECOND OCCURRENCE OF REPEATED VALUE
C        ...
C        N      NTH OCCURRENCE OF REPEATED VALUE
C     NOTE THAT IT IS ASSUMED THAT THE VALUES HAVE PREVIOUSLY
C     BEEN SORTED IN ASCENDING ORDER.
C     A SECOND ARRAY, INVRPT, IS ALSO GENERATED IN WHICH THE
C     VALUES ARE:-
C        N      NTH OCCURRENCE OF REPEATED FACTOR
C        ...
C        2      SECOND OCCURRENCE OF REPEATED FACTOR
C        1      SINGLE OR FIRST OCCURRENCE OF THE FACTOR
C     THE ARRAY INVRPT IS DERIVED FROM NREPT IN THE SECOND
C     PART OF THE SUBROUTINE
C
      INTEGER I,ILOW,IHIGH,INVRPT(1),NFACT,NREPT(1)
      COMPLEX DFACTS(1)
      LOGICAL MATCH
C
      MATCH=.FALSE.
```

```
C
      DO 10 I=1,NFACT-1
C
      IF ((DFACTS(I) .EQ. DFACTS(I+1)) .AND. .NOT. MATCH) THEN
C
        MATCH=.TRUE.
        NREPT(I)=1
        NREPT(I+1)=2
C
      ELSE IF ((DFACTS(I) .EQ. DFACTS(I+1)) .AND. MATCH) THEN
C
        NREPT(I+1)=NREPT(I)+1
C
      ELSE
C
        MATCH=.FALSE.
C
      END IF
C
   10 CONTINUE
C
C     THIS SECTION TAKES THE ARRAY NREPT THAT ACCOMPANIES
C     DFACTS, THE ARRAY OF DENOMINATOR FACTORS, AND WRITES
C     ANOTHER INTEGER ARRAY IN WHICH THE GROUPS OF REPEATED
C     FACTORS ARE FLAGGED IN DESCENDING ORDER.  AS AN
C     EXAMPLE:-
C     DFACTS  3.2  5.6  6.7  6.7  6.7  8.4  9.3  9.3  4.1
C     NREPT     0    0    1    2    3    0    1    2    0
C     INVRPT    1    1    3    2    1    1    2    1    1
C
C     THE ARRAY INVRPT INFORMATION WILL BE USED WHEN EACH
C     INDIVIDUAL PARTIAL FRACTION IS INVERTED BECAUSE
C     FRACTIONS OF THE FORM
C     1/(s+a)**n INVERT TO t**(n-1) exp(-at)/(n-1)!
C
C     THE OUTER LOOP OF THE FOLLOWING SET OF NESTED IF-THEN-
C     ELSE LOOPS IS SET UP AS  A DO-WHILE.  A SIMPLE DO LOOP
C     IS NOT USED BECAUSE THE LOOP COUNTING VARIABLE I IS NOT
C     INCREMENTED CONSISTENTLY.  INCREMENTATION IS UNIFORM FOR
C     NON-REPEATED FACTORS, BUT GROUPS OF REPEATED FACTORS ARE
C     TREATED AS GROUPS WITHIN INTERNAL  IF - THEN - ELSE
C     AND DO LOOPS.  THE LABEL 20 IS THE RETURN TARGET FOR
C     THE OUTER IF-THEN-ELSE.
C
      I=1
C
   20 CONTINUE
C
      IF (I .GT. NFACT) THEN
C
```

```
      ELSE
C
        IF (NREPT(I) .EQ. 0) THEN
C
        INVRPT(I)=1
        I=I+1
C
        ELSE
C
        ILOW=I
C
   30   CONTINUE
C
        I=I+1
C
          IF (NREPT(I) .NE. 0) THEN
C
          IHIGH=I
          GO TO 30
C
          ELSE
C
          DO 40 J=0,(IHIGH-ILOW)
C
          INVRPT(IHIGH-J)=NREPT(ILOW+J)
C
   40     CONTINUE
C
          END IF
C
        END IF
C
      GO TO 20
C
      END IF
C
      RETURN
      END

C...V....V....V....V....V....V....V....V....V....V....V....V

      SUBROUTINE PRTFRC (NNUMR,PNUMR,NDENM,DFACTS,J,CFF)
C
C     GLOSSARY OF VARIABLES
C     ---------------------
C
C     INTEGER
C     I          GENERAL COUNTING VARIABLE
C     J          ORDER OF THE PARTIAL FRACTION COEFFICIENT
                 BEING FOUND
```

```
C     NDENM      ORDER OF THE DENOMINATOR
C     NNUMR      ORDER OF THE NUMERATOR
C
C     REAL
C     PNUMR()    ARRAY OF COEFFICIENTS OF THE NUMERATOR COEFFI-
C                CIENTS, PNUMR(1) IS THE CONSTANT TERM
C
C     COMPLEX
C     CFF( )     ARRAY OF PARTIAL FRACTION COEFFICIENTS
C     DFACTS( )  ARRAY OF DENOMINATOR POLYNOMIAL FACTORS, THESE
C                ARE IN THE FORM a FROM (s+a)
C
C     SUBROUTINES OF FUNCTIONS CALLED
C     -------------------------------
C     CPEVAL( , , )      COMPLEX FUNCTION THAT EVALUATES A
C                        POLYNOMIAL AT A SPECIFIED VALUE
C
C     ALGORITHM DESCRIPTION
C     ---------------------
C     THIS ALGORITHM EVALUATES THE PARTIAL FRACTION COEFFI-
C     CIENTS FOR THE EXPANSION OF N(S)/D(S) WHEN D(S) HAS NO
C     REPEATED ROOTS.  FOR A DISCUSSION OF THE METHOD SEE
C     P. 700:- PALM, W. J. III.  1983.  MODELING, ANALYSIS,
C     AND CONTROL OF DYNAMIC SYSTEMS.  JOHN WILEY & SONS,
C     NEW YORK.

      INTEGER I,J,NDENM,NNUMR
      REAL PNUMR(21)
      COMPLEX CFF(NDENM),CPEVAL,DFACTS(NDENM),DROOT
C
      DROOT=-DFACTS(J)
      CFF(J)=CPEVAL(NNUMR,PNUMR,DROOT)
C
        DO 20 I=1,NDENM
C
          IF (I .NE. J) THEN
C
          CFF(J)=CFF(J)/(DROOT+DFACTS(I))
C
          ELSE
C
          CFF(J)=CFF(J)
C
          END IF
C
   20   CONTINUE
C
      RETURN
      END
```

```
C...V....V....V....V....V....V....V....V....V....V....V....V

      SUBROUTINE FCTPLY (NFACT,DFACTS,NOUT,POUT,CTEMP)
C
C     GLOSSARY OF VARIABLES
C     ---------------------
C     INTEGER
C     I          LOOP COUNTING VARIABLE
C     J          LOOP COUNTING VARIABLE
C     NFACT      NUMBER OF FACTORS
C     NOUT       ORDER OF THE RESULTING POLYNOMIAL
C
C     REAL
C     POUT()     ARRAY OF COEFFICIENTS OF THE RESULTING POLY.
C
C     COMPLEX
C     CTEMP( )   WORKING ARRAY OF THE RESULTING POLYNOMIAL
C                COEFFICIENTS IN COMPLEX FORM.
C     CTMP1      TEMPORARY VARIABLE
C     CTMP2      TEMPORARY VARIABLE
C     DFACTS()   ARRAY OF FACTORS THAT ARE BEING EXPANDED TO
C                FORM A POLYNOMIAL
C
C     ALGORITHM DESCRIPTION
C     ---------------------
C     THIS SUBROUTINE FINDS THE COEFFICIENTS OF A POLYNOMIAL
C     WHEN SUPPLIED WITH THE FACTORS.  THE POLYNOMIAL FORMED
C     HAS ALL REAL COEFFICIENTS SO ANY COMPLEX ROOTS MUST OCCUR
C     ONLY AS CONJUGATE PAIRS.
C     *** NOTE ***
C     POUT(1) IS THE CONSTANT TERM OF THE POLYNOMIAL
C
C     METHOD OF CALCULATING THE COEFFICIENTS IS AS FOLLOWS:
C
C     S**0                S**1     S**2            S**3          I
C     (S+A1)=             A1       1                             1
C                        =======================================
C     (S+A1)(S+A2)=       A1       A2                            2
C                                  A1              1
C                         --------------------------------------
C                         A1A2    (A1+A2)          1
C                         ======================================
C     (S+A1)(S+A2)(S+A3)= A1A2A3   A3(A1+A2)       A3            3
C                                  A1A2           (A1+A2)        1
C                         --------------------------------------
C                         A1A2A3  (A1A2+A2A3+A3A1)  (A1+A2+A3)   1
C                         ======================================
C
C     OBSERVE THAT THE COEFFICIENT OF CTEMP(J) CONSISTS OF TWO
C     TERMS, THE JTH COEFFICIENT ASSOCIATED WITH THE (I-1)TH
```

```
C     FACTOR MULTIPLIED BY THE ITH FACTOR AND CTEMP(J-1) ASSO-
C     CIATED WITH THE (I-1)TH FACTOR.  THIS SECOND TERM IS
C     OBTAINED BY COPYING CTEMP(J) TO A TEMPORARY VARIABLE
C     CTEMP2 IMMEDIATELY INSIDE THE J DO LOOP.
C     ANOTHER TEMPORARY VARIABLE CTEMP1 IS USED TO COPY CTEMP2
C     JUST BEFORE PERFORMING ANOTHER J ITERATION.  THE EXPRES-
C     SION FOR CTEMP(J) THEN BECOMES:-
C        CTEMP(J)=CTEMP(J)*DFACTS(I)+CTEMP1
C     WHERE CTEMP1 NOW HOLDS THE VALUE OF CTEMP(J-1) FOR THE
C     (I-1)TH FACTOR.  THE CONSTANT TERM CTEMP(1) IS HANDLED
C     SEPARATELY INSIDE THE I DO LOOP WITH THE EXPRESSION:-
C        CTEMP(1)=CTEMP(1)*DFACTS(I)
C     THIS ALGORITHM IS DESIGNED FOR USE WITH FACTORS FOR
C     WHICH COMPLEX VALUES ONLY APPEAR AS CONJUGATES, THUS THE
C     LAST STEP CONSISTS OF TAKING THE REAL PART OF CTEMP(I)
C     AND COPYING IT INTO POUT(I).
C     IN THEORY THE IMAGINARY PART OF CTEMP(I) SHOULD BE ZERO,
C     IN PRACTICE THERE MAY BE SOME SMALL RESIDUAL BECAUSE OF
C     FINITE ARITHMETIC INACCURACIES, BUT THIS IS IGNORED.
C
      INTEGER I,J,NFACT,NOUT
      REAL POUT(1)
      COMPLEX CTEMP(1),CTMP1,CTMP2,DFACTS(1)
C
      CTEMP(1)=(1.0,0.0)
C
      DO 10 I=2,(NFACT+1)
C
      CTEMP(I)=(0.0,0.0)
C
   10 CONTINUE
C
      DO 20 I=1,NFACT
C
      CTMP1=CTEMP(1)
      CTEMP(1)=CTEMP(1)*DFACTS(I)
C
        DO 30 J=2,(NFACT+1)
C
        CTMP2=CTEMP(J)
        CTEMP(J)=CTEMP(J)*DFACTS(I)+CTMP1
        CTMP1=CTMP2
C
   30   CONTINUE
C
   20 CONTINUE
C
      NOUT=NFACT
C
      DO 40 I=1,(NFACT+1)
```

```
C
      POUT(I)=REAL(CTEMP(I))
C
   40 CONTINUE
C
      RETURN
      END

C...V....V....V....V....V....V....V....V....V....V....V....V

      SUBROUTINE DFQUOT (NINN,PINN,NIND,PIND,NOUTN,POUTN,
     &                           NOUTD,POUTD,NTEMP,PTEMP)
C
C     GLOSSARY OF VARIABLES
C     ---------------------
C
C     INTEGER
C     NIND       ORDER OF THE ORIGINAL DENOMINATOR POLYNOMIAL
C     NINN       ORDER OF THE ORIGINAL NUMERATOR POLYNOMIAL
C     NOUTD      ORDER OF THE DENOMINATOR OF THE DIFFERENTIATED
C                POLYNOMIAL
C     NOUTN      ORDER OF THE NUMERATOR OF THE DIFFERENTIATED
C                POLYNOMIAL
C     NTEMP      ORDER OF THE TEMPORARY STORAGE POLYNOMIAL
C
C     REAL
C     PIND( )    COEFFICIENTS OF THE DENOMINATOR POLYNOMIAL
C     PINN( )    COEFFICIENTS OF THE NUMERATOR POLYNOMIAL
C     POUTD( )   COEFFICIENTS OF THE DENOMINATOR OF THE
C                DIFFERENTIATED FUNCTION
C     POUTN( )   COEFFICIENTS OF THE NUMERATOR OF THE
C                DIFFERENTIATED FUNCTION
C     PTEMP( )   COEFFICIENTS OF THE TEMPORARY STORAGE POLYNOMIAL
C         *** NOTE *** THIS IS A WORKING ARRAY AND MUST BE
C              PASSED FROM THE CALLING PROGRAMME
C
C     ALGORITHM DESCRIPTION
C     ---------------------
C     THIS SUBROUTINE ACCEPTS COEFFICIENT ARRAYS OF A NUMERATOR
C     POLYNOMIAL AND A DENOMINATOR POLYNOMIAL.  THE RULE FOR
C     EVALUATING THE FIRST DERIVATIVE OF A PRODUCT IS APPLIED
C     AND COEFFICIENT ARRAYS OF THE NUMERATOR AND DENOMINATOR
C     POLYNOMIALS OF THE DERIVATIVE ARE PASSED AS POUTN AND
C     POUTD.  ALTHOUGH THE EVALUATION PROCEDURE IS STANDARD,
C     POUTD, POUTN, AND PTEMP ARE USED AS INTERMEDIATE STORAGE
C     ARRAYS.  WITH THIS STRATEGY, ONLY ONE WORKING ARRAY IS
C     NEEDED BY THIS SUBROUTINE.  NOTE THAT THE SUBROUTINE
C     DFFPLY MUST BE ABLE TO PERFORM (A-B) -> A, WHERE
C     OLD A IS OVERWRITTEN WITH NEW A.  DFFPLY SATISFIES THIS
C     NEED.
```

```
C     SUBROUTINES NEEDED WITH THIS SUBROUTINE:-
C        MLTPLY    MULTIPLIES TWO POLYNOMIALS
         DFFPLY    FINDS THE DIFFERENCE BETWEEN TWO
C                  POLYNOMIALS.
C        DFPOLY    DIFFERENTIATES A POLYNOMIAL
C
      INTEGER NIND,NINN,NOUTD,NOUTN,NTEMP
      REAL PIND(1),PINN(1),POUTD(1),POUTN(1),PTEMP(1)
C
C     EVALUATE THE COEFFICIENTS OF U'
C
      CALL DFPOLY (NINN,PINN,NTEMP,PTEMP)
C
C     EVALUATE THE COEFFICIENTS OF U'V
C
      CALL MLTPLY (NTEMP,PTEMP,NIND,PIND,NOUTN,POUTN)
C
C     EVALUATE THE COEFFICIENTS OF V'
C
      CALL DFPOLY (NIND,PIND,NTEMP,PTEMP)
C
C     EVALUATE THE COEFFICIENTS OF UV'
C
      CALL MLTPLY (NINN,PINN,NTEMP,PTEMP,NOUTD,POUTD)
C
C     EVALUATE THE COEFFICIENTS OF U'V -UV'
C
      CALL DFFPLY (NOUTN,POUTN,NOUTD,POUTD,NOUTN,POUTN)
C
C     EVALUATE THE COEFFICIENTS OF V**2
C
      CALL MLTPLY (NIND,PIND,NIND,PIND,NOUTD,POUTD)
C
      RETURN
      END

C...V....V....V....V....V....V....V....V....V....V....V....V

       SUBROUTINE REVERSE ( NDENM, PDENM )

C  GLOSSARY OF VARIABLES
C  ---------------------

C    INTEGER

C      I         DO VARIABLE
C      J         COUNTING VARIABLE
C      K         CONSTANT
C      NDENM     THE ORDER OF REAL POLYNOMIAL D(S)
```

```
C    DOUBLE PRECISION

C       PDENM( ) THE COEFFICIENT ARRAY OF D(S)
C       TEMP     THE TEMPORARY STORAGE VARIABLE

C  ALGORITHM DESCRIPTION
C  ---------------------

C    THIS SUBROUTINE ACCEPTS THE COEFFICIENT ARRAY PDENM( )
C    IN ASCENDING ORDER AND RETURNS THE REVERSED PDENM( )
C    IN PLACE OF THE ORIGINAL PDENM( ).
C    ______________________________________________________

      INTEGER  I, J, K, NDENM
      DOUBLE PRECISION   PDENM(1), TEMP

C    REVERSE THE ORDER OF ARRAY PDENM( )

      K=(NDENM+1)/2
      DO 10 I=1, K
         TEMP=PDENM(I)
         J=(NDENM+1)-I+1
         PDENM(I)=PDENM(J)
         PDENM(J)=TEMP
10    CONTINUE

      RETURN
      END

C...V....V....V....V....V....V....V....V....V....V....V....V

      SUBROUTINE RESPONSE(CFF,DFACTS,INVRPT,NSUB,NFACT,
     $                                        TIME,Y,T)


C  GLOSSARY OF VARIABLES
C  ---------------------

C    INTEGER

C    I           DO VARIABLE
C    J           COUNTING VARIABLE
C    K           DO VARIABLE
C    INVRPT( )   INTEGER ARRAY COUNTING THE NUMBER OF REPEATED
C                  VALUES OF A MEMBER OF FACTORS
C    NFACT       ORDER OF REAL POLYNOMIAL D(S)
C    NSUB        THE NUMBER OF TIME SUBINTERVALS

C    REAL
```

```
C    TIME         TOTAL TIME LENGTH
C    T( )         ARRAY OF TIME VARIABLES
C    Y( )         THE TIME RESPONSE OF A LINEAR SYSTEM

C    COMPLEX

C    CFF( )       ARRAY OF PARTIAL FRACTION COEFFICIENTS
C    Y1( )        ANY TERM IN Y( ) FOR DISTINCT FACTOR OF D(S)
C    Y2( )        ANY TERM IN Y( ) FOR REPEATED FACTORS OF D(S)

C  ALGORITHM DESCRIPTION
C  ---------------------

C    THIS SUBROUTINE ACCEPTS THE COEFFICIENTS CFF( ), FACTORS
C    DFACTS( ) AND TIME ARRAY T( ) AND RETURNS THE VALUE OF
C    TIME RESPONSE ARRAY Y( ) FOR EACH TIME().

C  -----------------------------------------------------------

      INTEGER  I, J, K, INVRPT(1), NSUB, NFACT
      REAL     H, T(1), Y(1), TIME
      COMPLEX  CFF(1), DFACTS(1), SUM, Y1(10), Y2(10)

      DO 1000 K=1, NSUB

         T(K)=TIME*K/NSUB

C        INITIALIZATION OF VARIABLES

         DO 10 I=1, NFACT
            Y1(I)=(0.0,0.0)
            Y2(I)=(0.0,0.0)
10       CONTINUE

C        COMPUTE EACH TERM OF TIME RESPONSE Y( )

         J=0

100      CONTINUE

         J=J+1

         IF (INVRPT(J).EQ.1) THEN

C           RESPONSE TERM FOR DISTINCT FACTOR :

            Y1(J)=CFF(J)*CEXP(-DFACTS(J)*T(K))

         ELSE
```

```
C             RESPONSE TERM FOR REPEATED FACTORS :

              H=1.0
              DO 20 I=INVRPT(J)+(J-1), J, -1
                 H=H/(INVRPT(J)+J-I)
                 Y2(J)=Y2(J)+CFF(I)*H*(T(K)**(INVRPT(J)+J-I))*
     $                 CEXP(-DFACTS(J)*T(K))
20            CONTINUE

              J=INVRPT(J)+J-1

           ENDIF

           IF ( J.LT.NFACT ) GOTO 100

C  CALCULATE TIME RESPONSE Y( )

           SUM=(0.0,0.0)

           DO 30 I=1,NFACT
30            SUM=SUM+Y1(I)+Y2(I)

           Y(K)=REAL(SUM)

1000     CONTINUE

         RETURN

         END

C...V....V....V....V....V....V....V....V....V....V....V....V

        SUBROUTINE RESPNS(CFF,DFACTS,INVRPT,NSUB,NFACT,TIME,Y)


C  GLOSSARY OF VARIABLES
C  ---------------------

C    INTEGER

C    I          DO VARIABLE
C    J          COUNTING VARIABLE
C    K          DO VARIABLE
C    INVRPT( )  INTEGER ARRAY COUNTING THE NUMBER OF REPEATED
C                 VALUES OF A MEMBER OF FACTORS
C    NFACT      ORDER OF REAL POLYNOMIAL D(S)
C    NSUB       THE NUMBER OF TIME SUBINTERVALS
```

```
C    REAL

C    TIME         TOTAL TIME LENGTH
C    Y( )         THE TIME RESPONSE OF A LINEAR SYSTEM

C    COMPLEX

C    CFF( )       ARRAY OF PARTIAL FRACTION COEFFICIENTS
C    Y1(), Z1()     ANY TERM IN Y( ) FOR DISTINCT FACTOR OF D(S)
C    Y2(), Z2()     ANY TERM IN Y( ) FOR REPEATED FACTORS OF D(S)

C  ALGORITHM DESCRIPTION
C  ---------------------

C    THIS SUBROUTINE ACCEPTS THE COEFFICIENTS CFF( ), FACTORS
C    DFACTS( ) AND TIME ARRAY T( ) AND RETURNS THE VALUE OF
C    TIME RESPONSE ARRAY Y( ) FOR EACH TIME().

C    ---------------------------------------------------------

      INTEGER  I, J, K, INVRPT(1), NSUB, NFACT
      REAL     H, TIME, Y(1)
      COMPLEX  CFF(1), DFACTS(1), SUM, SUM1, Y1(10), Y2(10)
      COMPLEX  Z1(10), Z2(10)

      DO 1000 K=1, NSUB


C        INITIALIZATION OF VARIABLES

         DO 10 I=1, NFACT
            Y1(I)=(0.0,0.0)
            Y2(I)=(0.0,0.0)
            Z1(I)=(0.0,0.0)
            Z2(I)=(0.0,0.0)
10       CONTINUE

C        COMPUTE EACH TERM OF TIME RESPONSE Y( )

         J=0

100      CONTINUE

         J=J+1

         IF (INVRPT(J).EQ.1) THEN

C           RESPONSE TERM FOR DISTINCT FACTOR :

            Y1(J)=CFF(J)*CEXP(-DFACTS(J)*(TIME/NSUB)*K)
```

```
            Z1(J)=CFF(J)*CEXP(-DFACTS(J)*(TIME/NSUB)*(K-1))

         ELSE

C           RESPONSE TERM FOR REPEATED FACTORS :

            H=1.0
            DO 20 I=INVRPT(J)+(J-1), J, -1
               H=H/(INVRPT(J)+J-I)
               Y2(J)=Y2(J)+CFF(I)*H*((TIME/NSUB)*K)**
     $               (INVRPT(J)+J-I)*
     $               CEXP(-DFACTS(J)*(TIME/NSUB)*K)
               Z2(J)=Z2(J)+CFF(I)*H*(TIME/NSUB)*(K-1)**
     $               (INVRPT(J)+J-I)*CEXP(-DFACTS(J)*
     $               (TIME/NSUB)*(K-1))
20          CONTINUE

            J=INVRPT(J)+J-1

         ENDIF

         IF ( J.LT.NFACT ) GOTO 100

C  CALCULATE TIME RESPONSE Y( )

         SUM=(0.0,0.0)
         SUM1=(0.0,0.0)

         DO 30 I=1,NFACT
            SUM=SUM+Y1(I)+Y2(I)
            SUM1=SUM1+Z1(I)+Z2(I)
30       CONTINUE

         IF(K.EQ.1) THEN
           Y(K)=REAL(SUM)
         ELSE
           Y(K)=REAL(SUM)-REAL(SUM1)
         ENDIF

1000    CONTINUE

        RETURN

        END

C...V....V....V....V....V....V....V....V....V....V....V....V

      SUBROUTINE SIMPSON( NSUB, TIME, T, Y, S )

C   ALGORITHM DESCRIPTION
```

```
C     THE COMPOSITE SIMPSON'S RULE IS UTILIZED FOR THE
C     ESTIMATION OF THE AREA UNDER A CURVE LINE.

C   INPUT***

C     NSUB   THE NUMBER OF TIME SUBINTERVALS
C     TIME   THE MAXIMUM VALUE OF TIME VARIABLE T( )
C     T( )   TIME VARIABLE ARRAY
C     Y( )   TIME RESPONSE OF A LINEAR SYSTEM

C   OUTPUT***

C     S      THE ESTIMATION OF THE AREA

C   INTERMEDIATE VARIABLES***

C     S0     SUMMATION OF THE BOTH END VALUES OF INTEGRAND
C     S1     SUMMATION OF THE ODD TERMS OF INTEGRAND VALUES
C     S2     SUMMATION OF THE EVEN TERMS OF INTEGRAND VALUES
C     H      THE VALUE OF EACH EQUALLY SPACED SUBINTERVAL

C     ******* NOTE *******

C     SUPPOSE THAT BOTH INTEGRAND AND INDEPENDENT VARIABLES
C     START FROM ZERO.

C     ________________________________________________________

      INTEGER  I, NSUB
      REAL     H, S, S1, S2, TIME, T(1), Y(1)

      H=TIME/NSUB
      S0=T(NSUB)*Y(NSUB)**2
      S1=0.
      S2=0.

      DO 10 I=1,NSUB-1,2
10       S1=S1+T(I)*Y(I)**2

      DO 20 I=2,NSUB-1,2
20       S2=S2+T(I)*Y(I)**2

      S=H*(S0+4*S1+2*S2)/3.

      RETURN

      END
```

APPENDIX B

```
C                  PROGRAM  FCNMIN.FOR

C     THIS PROGRAM ATTEMPTS TO MINIMIZE A FUNCTION
C     SPECIFIED AS AN EXTERNAL FUNCTION BY THE USER.
C                         *****  NOTE  *****
C     IF THE NAME OF THE EXTERNAL FUNCTION IS NOT FCN, THEN
C     A GLOBAL CHANGE SHOULD BE PERFORMED TO CHANGE THE NAME
C     FCN TO WHATEVER NAME HAS ACTUALLY BEEN USED FOR THE
C     FUNCTION BEING MINIMIZED.
C     NOTE ALSO THAT THE VALUE NX=4 IN THE PARAMETER STATEMENT
C     SHOULD BE CHANGED TO CORRESPOND TO THE NUMBER OF
C     VARIABLES IN THE FUNCTION BEING MINIMIZED.
C                         *****  ****  *****
C     THE MINIMIZATION SUBROUTINES DEVELOPED BY:
C     DENNIS, J. E. JR. AND SCHNABEL, R. B.  1983.  NUMERICAL
C     METHODS FOR UNCONSTRAINED OPTIMIZATION AND NONLINEAR
C     EQUATIONS.  PRENTICE HALL, ENGLEWOOD CLIFFS, NEW JERSEY.
C     ARE USED.  THE BFGS SECANT UPDATE IS USED IN THE
C     UNFACTORED FORM.  PLEASE NOTE THAT THE LINE SEARCH IS
C     A SIMPLE LINE SEARCH AND NOT THE MODIFIED ALGORITHM
C     RECOMMENDED BY DENNIS AND SCHNABEL.
C     THE ALGORITHM IS INTERACTIVE AND PERFORMS ONE ATTEMPTED
C     IMPROVEMENT OF THE VARIABLE VECTOR BEFORE REQUESTING
C     PERMISSION TO CONTINUE.
C
C     THE DENNIS AND SCHABEL SUBROUTINES USED ARE:-
C     FDGRAD, INTHSS, BFGSUF, CHSOLV, CHDCMP, LSOLV, LTSOLV,
C     LINSRC,MCHEPS,UMSTOP
C
      EXTERNAL FCN
C
      INTEGER I,ITNCNT,ITNLMT,NCNSCM,NRETN,NTERM,NX
      PARAMETER (NX=4)
      DOUBLE PRECISION XC(NX),FCXC,H(NX,NX),DL(NX,NX),
     $DMEPS,ETA,SX(NX),FCN,GRADXC(NX),GRDTOL,GRPLUS(NX),
     $S(NX),DMXSTP,STPTOL,XPLUS(NX),FPLUS,TYPF,DMADD,DMOFFL
      LOGICAL ANLGRD,MAXTKN
      CHARACTER*1 Q
C
      WRITE (*,*) 'UNCONSTRAINED MINIMIZATION ROUTINE TAKEN
     $ FROM DENNIS AND SCHNABEL'
      WRITE (*,*) ' '
      WRITE (*,*) 'ENTER ESTIMATES OF THE VARIABLES AT
                   THE MINIMUM'
      DO 10 I=1,NX
C
      WRITE (*,200) I
```

```
  200 FORMAT (1X,'ENTER X(',I2,') AS X.XXD+YY ')
      READ (*,*) XC(I)
      WRITE (*,300) I
  300 FORMAT (1X,'ENTER SX(',I2,') AS X.XXD+YY NOTE:
     $         Sx=|1/typx|')
      READ (*,*) SX(I)
C
   10 CONTINUE
C
      ANLGRD=.FALSE.
      WRITE (*,*) 'ENTER TYPF AS X.XXD+YY '
      READ (*,*) TYPF
      WRITE (*,*) 'ENTER DMXSTP AS X.XXD+YY   '
      READ (*,*) DMXSTP
C
      WRITE (*,*) ' '
C
      CALL MCHEPS (DMEPS)
      WRITE (*,*) 'DMEPS= ',DMEPS
      ETA=1.0D-10
      GRDTOL=DMEPS**0.67
      STPTOL=DMEPS**0.67
C
      ITNLMT=50
      WRITE (*,*) ' '
C
      FCXC=FCN(NX,XC)
      CALL FDGRAD (NX,XC,FCXC,FCN,SX,ETA,GRADXC)
      CALL INTHSS (NX,FCXC,TYPF,SX,H)
      ITNCNT=0
C
C     THIS NEXT CONTINUE STATEMENT IS THE ENTRY POINT FOR
C     ANOTHER ITERATION.
C
  100 CONTINUE
C
      ITNCNT=ITNCNT+1
      WRITE (*,*) 'GRADIENT','       FCXC= ',FCXC
C
      WRITE (*,1000)  (GRADXC(I),I=1,NX)
 1000 FORMAT (6(E10.3,1X))
C
      WRITE (*,*) '    '
C
      WRITE (*,*) 'HESSIAN'
C
      DO 30 I=1,NX
C
      WRITE (*,1100) (H(I,K),K=1,NX)
 1100 FORMAT (8(E10.3,2X))
```

```
C
   30 CONTINUE
C
      WRITE (*,*) '   '
C
C     H IS KNOWN TO BE POSITIVE DEFINITE SO WE CAN SET
        DMOFFL=0.
      DMOFFL=0.0D0
      CALL CHDCMP (NX,H,DMOFFL,DMEPS,DL,DMADD)
      CALL CHSOLV (NX,GRADXC,DL,S)
C
      CALL LINSRC (NX,XC,FCXC,FCN,GRADXC,S,SX,DMXSTP,STPTOL,
     $NRETN,XPLUS,FPLUS,MAXTKN)
      WRITE (*,*) 'NRETN= ',NRETN,' MAXTKN= ',MAXTKN
      WRITE (*,*) '  '
C
C     SET UP THE VALUES FOR ANOTHER ITERATION USING A
C     SECANT UPDATE FOR THE HESSIAN.
C
      CALL FDGRAD (NX,XPLUS,FPLUS,FCN,SX,ETA,GRPLUS)
      CALL UMSTOP (NX,XC,XPLUS,FPLUS,GRPLUS,SX,TYPF,NRETN,
     $GRDTOL,STPTOL,ITNCNT,ITNLMT,MAXTKN,NCNSCM,NTERM)
      WRITE (*,*) 'ITNCNT= ',ITNCNT,' NCNSCM= ',NCNSCM,
     $             'NTERM= ',NTERM
      IF (NTERM .EQ. 0) THEN
C
      CALL BFGSUF (NX,XC,XPLUS,GRADXC,GRPLUS,DMEPS,ETA,
     $            ANLGRD,H)
      FCXC=FPLUS
C
      DO 50 I=1,NX
      XC(I)=XPLUS(I)
      WRITE (*,*) 'X(',I,')= ',XC(I)
      GRADXC(I)=GRPLUS(I)
   50 CONTINUE
C
  110 CONTINUE
C
      WRITE (*,*) 'ANOTHER ITERATION? Y OR N '
      READ (*,1200) Q
 1200 FORMAT (A1)
C
        IF ((Q .EQ. 'y') .OR. (Q .EQ. 'Y')) THEN
C
        GO TO 100
C
        ELSE IF ((Q .EQ. 'n') .OR. (Q .EQ. 'N')) THEN
C
        ELSE
C
```

```
          GO TO 110
C
          END IF
C
        ELSE
C
        WRITE (*,*) 'PROGRAM TERMINATED, A STOPPING
     $              CRITERION WAS'
        WRITE (*,*) 'REACHED IN THE SUBROUTINE UMSTOP'
C
        END IF
C
        STOP
        END
C
        DOUBLE PRECISION FUNCTION FCN (NX,X)
C
        INTEGER   NX
        DOUBLE PRECISION   X, PINDX

        CALL  PINDEX ( NX, X, PINDX )

        RETURN
        END
```

APPENDIX C

```
C           RANDOM NUMBER GENERATOR

C   ALGORITHM DESCRIPTION
C   THE PROGRAM GENERATES A SET OF RECTANGULARLY DISTRIBUTED
C     NUMBERS BETWEEN 0 AND 1, INCLUDING THE END POINTS

C   INPUT***
C     L         ANY INTEGER NUMBER

C   OUTPUT***
C     U(I,J)    TWO DIMENSIONAL ARRAY FOR THE RANDOM NUMBERS

C____________________________________________________________

      INTEGER   I, J, L
      REAL      RAND, U(100,100)

      COMMON /RAN/ IX, IY, IZ
      IX=14265
      IY=25276
      IZ=13129

      OPEN (UNIT=1, FILE='DATA', STATUS='NEW')

      WRITE(*,*) 'ENTER THE VALUE OF INTEGER VARIABLE: L'
      READ(*,*) L
      WRITE(1,*) 'RANDOM NUMBER TABLE'
      DO 10 I=1,64
         DO 10 J=1,8
            U(I,J)=RAND(L)
10    CONTINUE

      WRITE(1,222) (( U(I,J),J=1,8),I=1,64)
222   FORMAT(1X, 8F8.4)

      CLOSE(UNIT=1)

      STOP

      END

C____________________________________________________________

      FUNCTION RAND(L)

C   THE FUNCTION RETURNS A PSEUDO-RANDOM NUMBER RECTANGULARLY
```

```
C       DISTRIBUTED BETWEEN 0 AND 1.

C    IX, IY AND IZ SHOULD BE SET TO INTEGER VALUES BETWEEN  1
C       AND 30000 BEFORE FIRST ENTRY
      COMMON /RAN/ IX, IY, IZ

      IX=171*MOD(IX,177)-2*(IX/177)
      IY=172*MOD(IY,176)-35*(IY/176)
      IZ=170*MOD(IZ,178)-63*(IZ/178)

      IF (IX.LT.0) IX=IX+30269
      IF (IY.LT.0) IY=IY+30307
      IF (IZ.LT.0) IZ=IZ+30323

      RAND=AMOD(FLOAT(IX)/30269.0+FLOAT(IY)/30307.0+
     $          FLOAT(IZ)/30323.0,1.0)

      RETURN
      END
```

TABLE 2. A list of random lateral disturbing forces (kN)

0.1650	0.5073	0.0439	0.5666	0.8593	0.2996	0.2162
0.9078	0.5633	0.8255	0.3018	0.9820	0.5030	0.3681
0.6104	0.8832	0.5278	0.8878	0.3396	0.3817	0.8349
0.4071	0.2816	0.8211	0.2673	0.5501	0.8300	0.9738
0.3813	0.2807	0.7067	0.3577	0.4336	0.0858	0.7255
0.5512	0.7871	0.7063	0.2122	0.5979	0.4715	0.6279
0.4301	0.8754	0.6601	0.7773	0.4192	0.4059	0.2693
0.4173	0.0640	0.6672	0.3417	0.0744	0.8275	0.4508
0.5798	0.5912	0.4422	0.5658	0.6868	0.7270	0.9446
0.1516	0.1927	0.9179	0.0701	0.6764	0.6339	0.7665
0.2016	0.0911	0.5428	0.0702	0.5942	0.9820	0.8925
0.0453	0.3970	0.0757	0.5446	0.4984	0.4470	0.3774
0.5326	0.4690	0.9521	0.4323	0.8744	0.3598	0.1198
0.8041	0.1197	0.6694	0.0041	0.1564	0.9129	0.3042
0.3955	0.0646	0.9168	0.8320	0.0096	0.2661	0.3695
0.3142	0.4302	0.4281	0.3139	0.6848	0.1320	0.5245
0.8574	0.8967	0.5108	0.2340	0.4523	0.7809	0.8679
0.3045	0.7310	0.9519	0.9817	0.8277	0.2297	0.2855
0.9980	0.6273	0.7639	0.4487	0.5995	0.1172	0.2623
0.0321	0.4770	0.1397	0.0352	0.2170	0.3109	0.6013
0.5987	0.4350	0.2217	0.0196	0.9263	0.2080	0.7781
0.1987	0.3553	0.5582	0.6541	0.0582	0.6374	0.9751
0.5880	0.9035	0.6611	0.5617	0.5149	0.7878	0.2459
0.8132	0.4766	0.9413	0.0659	0.2081	0.5854	0.4869
0.2322	0.6378	0.2906	0.0389	0.1509	0.6127	0.3730
0.2927	0.0666	0.3693	0.2814	0.0473	0.0634	0.9763
0.9650	0.3942	0.9551	0.5510	0.7116	0.2461	0.8951
0.0004	0.3455	0.5262	0.0256	0.9724	0.1638	0.8932
0.9274	0.1390	0.3701	0.2187	0.8479	0.0159	0.7892
0.6681	0.6329	0.9581	0.5091	0.1562	0.0202	0.4608
0.0737	0.6109	0.7326	0.0345	0.7337	0.2039	0.8186
0.2947	0.2312	0.6080	0.9762	0.8225	0.5198	0.0275
0.8730	0.4094	0.4773	0.0649	0.2070	0.3762	0.8926
0.4509	0.8722	0.1772	0.1060	0.8636	0.2366	0.9086
0.7350	0.5106	0.1500	0.0589	0.2918	0.5647	0.6708
0.0520	0.2650	0.3942	0.3286	0.4421	0.7257	0.5523